DVD+RW Video Format Verifier

Release 1.6

User Manual

Graphical User Interface version

Document				
Author	:	Meindert Schuitema	Date:	2005-08-16
Reference	:	DSE-245023MH C09S2	Filename:	DVDRW_GUIUserManual.doc
Version	:	1.6.1	Archive:	Project / Dev. Support
Status	:	Approved	Classification:	--

DISCLAIMER

The information contained herein is believed to be accurate as of the date of publication, however Philips Electronics N.V. will not be liable far any damages, including indirect or consequential, resulting from the use of the software or reliance on the accuracy of this information. The information contained herein is subject to change without notice.

REPRODUCTION NOTICE

The software described in this document is intended to be used on a single computer system. Distribution of the software or documentation, whole or in part, to any other system or to any other party may constitute a misappropriation of trade secrets and confidential processes which are the property of Philips Consumer Electronics B.V. or other parties. Unauthorised distribution of software may cause damages far in excess of the value of the copies involved.

DOCUMENT CHANGE HISTORY

Date	Person	Version	Reason
2000-11-27	Jaya Hariharan	0.1	Draft
2001-03-08	Ofi	0.2	Updated
2001-07-23	Ofi	0.3	Final Draft
2001-07-25	Jaya Hariharan	0.4	added review comments
2001-07-26	Ofi	0.5	Added DVD+RW message list
2001-08-03	Ofi	0.6	Added Chapters 12, 13
2001-08-08	Jaya Hariharan	0.7	Reviewed Chapter DVD+RW message list
2001-10-22	Ofi	1.0	Approved release 1.0.0 version
2002-09-13	Maurice Hebben	1.1	Updated for release 1.1.0 version
2003-04-16	Maurice Hebben	1.2	Updated for release 1.2.0 version
2004-03-16	Maurice Hebben	1.3	Updated for release 1.3.1 version
2004-08-23	Maurice Hebben	1.4	Updated for release 1.4.0 version
2004-12-21	Maurice Hebben	1.5	Updated for release 1.5.0 version
2005-08-02	Meindert Schuitema	1.6	Updated for release 1.6.0 version
2005-08-16	Meindert Schuitema	1.6.1	Updated for release 1.6.1 version

TABLE OF CONTENTS

1 INTRODUCTION

The DVD+RW Video Format Verifier is a verification tool developed by the Philips Digital Systems Lab (formerly known as Philips Consumer Electronics - ASA Lab Eindhoven) for Philips Intellectual Properties & Standards (formerly known as Philips Systems Standards & Licensing) to support the standardisation process and the creation of DVD+RW Video discs and DVD+R Video discs, compliant with the DVD+RW/+R Basic Format Standard and DVD+RW/+R Video Format standard. The verifier performs syntax checks as well as semantic and dynamic checks, and cross checks the consistency between various data elements. The tool generates reports with all detected violations of the standards. Additionally, the input stream can be analysed and its contents on all specified levels can be logged in a clear and concise way.

1.1 PURPOSE

This manual explains how to use the DVD+RW Video Verifier. Furthermore, an overview of all verifier checks & error messages is given and some guidance on how to interpret these messages.

1.1 SCOPE

This document relates to version 1.6.1 / 4.8.0 of the DVD+RW Video Format Verifier for the only currently supported platforms, i.e. Windows 2000 and Windows XP. The version numbers reflect both the version of the DVD+RW specific part (1.6.0) and of the underlying verification core libraries (4.8.0). Furthermore, the DVD+RW Video Format Verifier checks the input bit streams against the currently most recent version of the DVD+RW Basic Specification v1.3 and DVD+RW Video Specification, which is v3.0.

This version 1.6.1 / 4.8.0 the DVD+RW Video Format verifier tool supports next to the friendly Graphical User interface also the functionality of the console version (useful for test scripts, i.e. batch runs).

Both are integrated in this executable of the verifier application.

1.2 DEFINITIONS, ACRONYMS & ABBREVIATIONS

Abbreviations:

ASA	Advanced Systems and Applications
AU	Access Unit
AVDP	Anchor Volume Descriptor Pointer
BEE	Basic Engine Emulator
BL	Bitrate Level
BSWE	Bad Spot Write Error
C_ADT	Cell Address Table
CBR	Constant Bit Rate
CMF	Cutting Mastering Format
CPSI	Copy Protection System Information
CVBR	Constrained Variable Bit Rate
DCB	Disc Control Block
DDP	Disc Description Protocol
DIZ	Disc Identification Zone
DVD	Digital Versatile Disc
DVD+RW	DVD-Rewritable
DVD+VR	DVD-Video Recording Format
ECC	Error Correction Code
FDCB	Formatting DCB
FP-PGC	First Play Program Chain
LSB	Least Significant Byte
LSN	Logical Sector Number
MP@ML	Main Profile at Main Level $\in$ MPEG
MPEG	Moving Pictures Expert Group
MSB	Most Significant Byte
N.A.	Not Applicable
PCI	Presentation Control Information
PCI_GI	PCI General Information
PFI	Physical Format Information
PS	Program Stream
PSN	Physical Sector Number
RLBN	Relative Logical Block Number
SPU	Sub Picture Unit
TTU	Title Unit
UDF	Universal Disk Format
UOP	User Operation
UTC	Universal Time
VBR	variable bit rate
VCPS	Video Content Protection System
VMG	Video Manager
VMGI	Video Manager Information
VMGM	Video Manager Menu
VMGM_VOBS	VMGM VOBS
VOB	Video Object
VOBs	Video Objects
VOBS	Video Object Set
VOBU	Video Object Unit
VRMI	Video Recording Manager Information

VRPL	Video Recording Play List
VTS	Video Title Set
VTS_C_ADT	VTS Cell Address Table ∈ VTSI
VTS_VOBU_ADMAP	VTS VOBU Address Map ∈ VTSI
VTSI	Video Title Set Information
VTSM	Video Title Set Menu
VTSs	Video Title Sets
VTSTT_VOBS	VTS Title VOBS

1.3 REFERENCES

The DVD+RW (Video) specifications are described in:

[DVD+RW] *DVD+RW 4.7 Gbytes Basic Format Specifications*
Hewlett-Packard, Mitsubishi Chemical, Philips, Ricoh, Sony, Yamaha
Version 1.3, July 2004

[DVD+VR] *DVD+RW Video Format Specifications*
Philips
Version 3.0, July 2005

[DVD+R] *DVD+R 4.7 Gbytes Basic Format Specifications*
Hewlett-Packard, Mitsubishi Chemical, Philips, Ricoh, Sony, Yamaha
Version 1.3, July 2004

[DVD+VRR] *DVD+R Video Format Specifications*
Philips
Version 1.2, June 2004

[VCPS] Video Content Protection System
for the DVD+R/+RW
Video Recording Format
Philips
Version 1.3, April 2005

The underlying standards are described in:

[MPEG-1 Video] *ISO/IEC 11172-2: 1993 Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s –*
Part 2: Video (MPEG-1 Video)

[MPEG-1 Audio] *ISO/IEC 11172-3: 1993 Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s –*
Part 3: Audio (MPEG-1 Audio)

[MPEG-2 Systems] *ISO/IEC 13818-1: Information technology –*
Generic coding of moving pictures and associated audio information:
Part 1: Systems (MPEG-2 Systems)

[MPEG-2 Video] *ISO/IEC 13818-2: Information technology –*
Generic coding of moving pictures and associated audio information:

Part 2: Video (MPEG-2 Video)

[MPEG-2 Audio] *ISO/IEC 13818-3 Second Edition: Information technology – Generic coding of moving pictures and associated audio information:*
Part 3: Audio (MPEG-2 Audio)

[DVD-PHYS] *DVD Specifications for Read-Only Disc – Part 1: Physical Specifications*
(Version 1.01, December 1997)

[DVD-FS] *DVD Specifications for Read-Only Disc – Part 2: File System Specifications*
(Version 1.01, December 1997)

[DVD-Video] *DVD Specifications for Read-Only Disc – Part 3: Video Specifications*
(Version 1.1, December 1997)

[ECMA] *Volume and File Structure for Write-Once and Rewritable Media using Non-Sequential Recording for Information Interchange*
Standard ECMA - 167,
3rd Edition – June 1997

[UDF] *Universal Disk Format Specification*
OSTA, Optical Storage Technology Association
Revision 1.02

[ISO] *Information processing – Volume and file structure of CD-ROM for information interchange*
International Standard, ISO 9660
First edition 1998-04-15, corrected 1988-09-01

[DDP] *DDP Specification*,
Doug Carson & Associates, August 4, 1998
Version 2.00

[CMF] *DVD Cutting Master Format Specification*,
23 September, 1999
Version 1.00

[AC-3] *ATSC Doc. A/52*, *Digital Audio Compression Standard (AC-3)*
20 Dec 95

EIA-608 1994 *Recommended practices for Line 21 data services*

ITU-R BT.601-5 1995 *Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios*

ETSI EN 300 294 *Television systems; 625-line television Wide Screen Signalling (WSS)*
V1.3.2, 1998-04

ETSI EN 300 468 *Digital Video Broadcasting (DVB);*
Specification for Service Information (SI) in DVB systems
V1.3.1, 1998-02

ISO/IEC 8859-1 *Information processing – 8-bit single-byte coded graphic character sets*

Part 1: Latin alphabet No.1

IEC 60958-1, 1999 *Digital Audio Interface – Part 1: General*

IEC 60958-3, 1999 *Digital Audio Interface – Part 3: Consumer Applications*

IEC 61937, 2000 *Digital audio – Interface for non-linear PCM encoded audio bitstreams applying IEC 60958*

2 VERIFIER PROPERTIES

2.1 FEATURES

This version of the DVD+RW Video Format Verifier supports parsing and verification of:

On MPEG level:

- Program Stream:
  - pack
  - system_header
  - PES_packet
  - PES_packet data
  - SCR timing
  - P-STD buffer model

- MPEG-1 and MPEG-2 MP@ML:
  - sequence_headers
  - GOP_header
  - picture
  - slice
  - macroblock
  - block
  - video VBV or Leak Method buffer model

- Complete parsing of MPEG video and MPEG audio layer I & II, incl. multi-channel audio, audio frames, headers and data

On DVD-Video level:

- UDF and ISO-9660 filesystem
- VMGI
- VTSI
- PGCI
- private_stream_1 data
- private_stream_2 data
- SPU
- PCI
- DSI
- Navigation Commands
- LPCM
- Dolby AC-3
- Cross-checks between VMGI, VTSI, PGCI and VOBS

On DVD+RW Video level:

- VRMI
- PCI_GI Extension data
- Cross-checks of DVD+RW data with DVD-Video and MPEG data

On DVD+RW physical level:

- Sector header
- Lead-in and Lead-out data

It accepts as input the following data streams:

- Files with DVD+RW compliant file naming conventions are supported and will be verified according the DVD+RW (Video) spec. These include: VIDEO_RM.IFO, VIDEO_RM.BUP, VIDEO_TS.IFO, VIDEO_TS.BUP, VIDEO_TS.VOB, VTS_<01..03>_0.IFO, VTS_<01..03>_0.BUP, VTS_<01..03>_<0..5>.VOB. These filenames may be preceded by any combination of characters.
- When verification of a disc is requested, only standard DVD+RW files normally present on a DVD+RW-disc are supported, i.e. IFO & their backup (.BUP) or VOB files. Verification of files in directories other than the /VIDEO_RM and /VIDEO_TS directory is not supported.
- Generic MPEG-2 Program streams are accepted, but must use a filename that contains no ".VOB",".IFO" or ".BUP" extension or "0_" prefix, otherwise they might be confused for DVD-Video or DVD+RW specific files.
- Files other than DVD(+RW) specific files, such as MPEG PES, MPEG Video or Audio ES, MPEG private_stream_1 or private_stream_2 or DVD(+RW) specific private data streams such as SPU or PCI are NOT supported. When presented to the verifier, files of this type will be incorrectly verified as a Program Streams!
- DDP disc images with the "0_"-prefix or "1_"-prefix file naming convention are supported. (The "1_"-prefix part of the disc image cannot be verified separately from the "0_"-prefix). This file name convention is used by Philips DVD authoring tools.
- The DVD DDPv2.0 disc images using the "DDP", "CONTROL.DAT" and "MAIN.DAT" file naming convention are supported.
- The DVD CMF disc images using the "DDVID.DAT", "CONTROL.DAT", "IMAGE.DAT" file naming convention are supported.
- The ISO disc images using .ISO extension are supported.
- Actual DVD+RW discs.
  Use a legacy DVD-ROM or DVD+RW drive or a special DVD verification drive described in Appendix B.
  Note: "Physical data" (i.e. sector headers, Lead-in or Lead-out) of actual discs can only be verified when using the special DVD verification drive as described in Appendix B.

2.2 PLATFORM

The current version of the tool runs under:

- **MS Windows NT™ v4.0** with Service Pack **6a** installed.
- **MS Windows 2000™** with Service Pack **2** (or higher) installed.
- **MS Windows XP™**

3 GRAPHICAL USER INTERFACE

3.1 OVERVIEW

Shown below is the main dialog of the DVD+RW GUI. This is shown to the user, once the software is registered successfully.

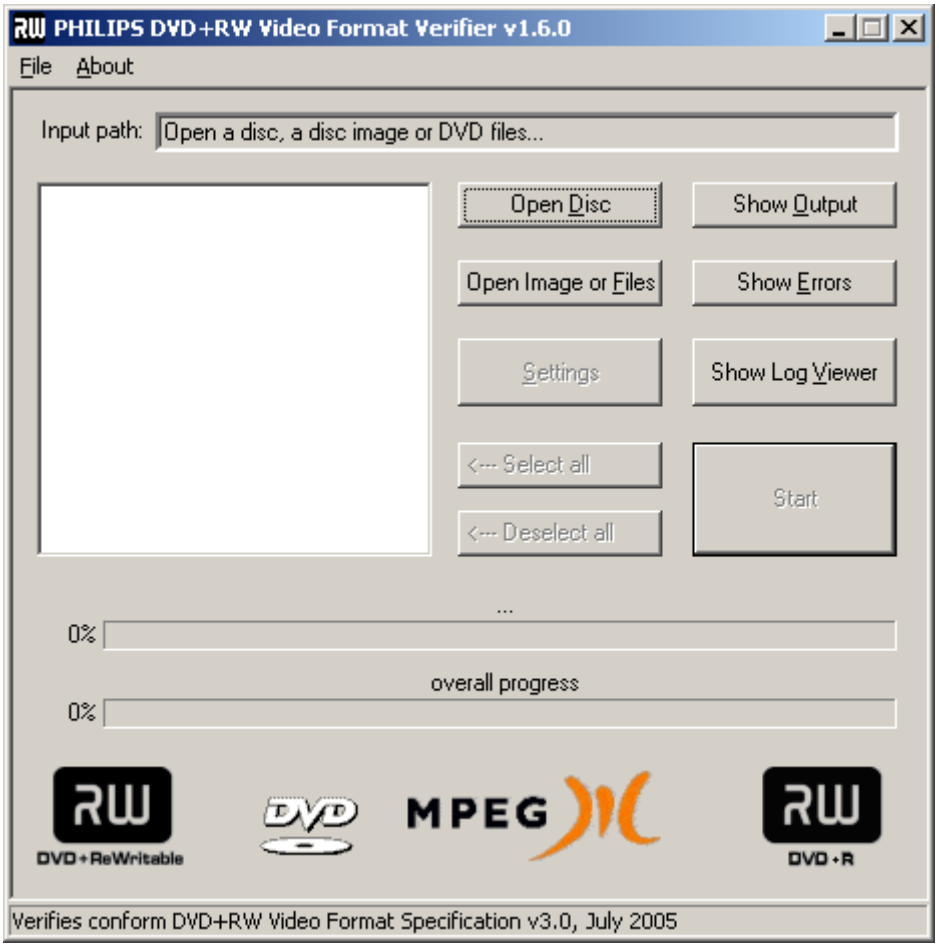


Figure 3-1 Main Dialog of DVD+RW GUI Verifier

On application start-up the following splash window is shown to the user.
This disappears after a short duration or if the user left-clicks on the splash window.

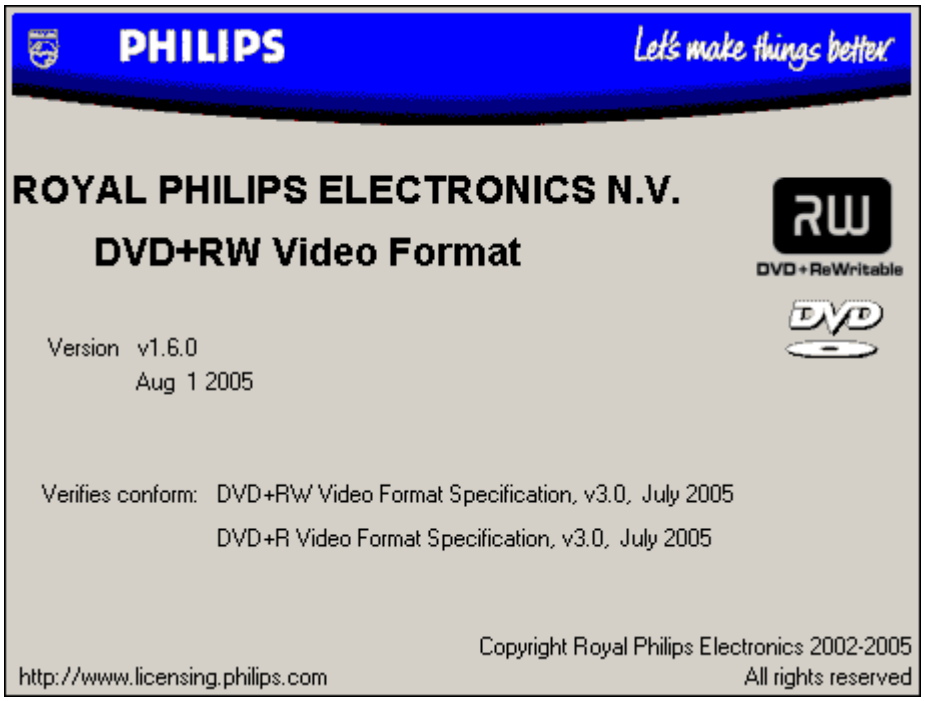


Figure 3-2: Splash window

3.2 REGISTRATION OF THE DVD+RW VERIFIER

Only copy-protected versions of the verifier have to be registered before first use.

This registration is “node-locked”: it allows use of the tool only on the system it has been registered on. It is possible to re-install the tool on another PC or platform, but it has to be registered again.

This section describes the registration scenario.

1. On start-up of the GUI, the following identification dialog is shown:

Figure 3-3: Identification dialog

2. The user is prompted with a dialog asking for either immediate registration or registration later. This dialog is shown in the figure below:

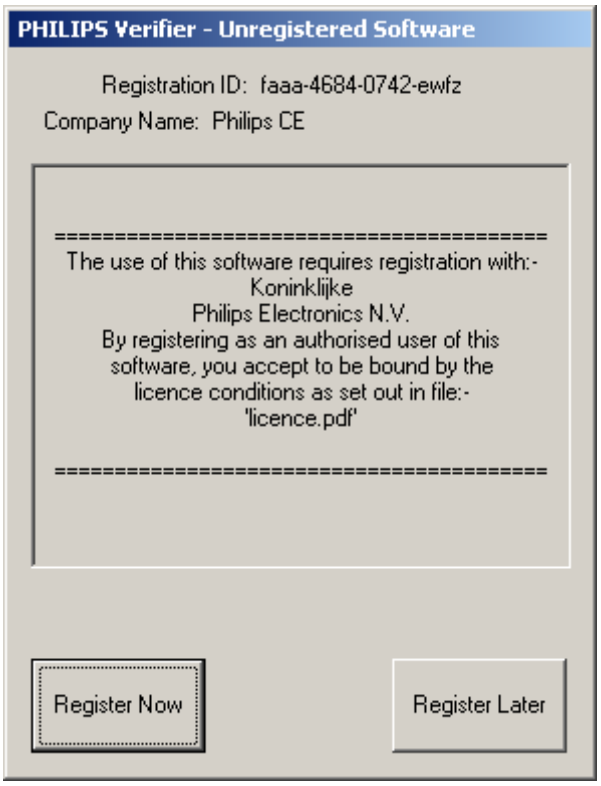


Figure 3-4: Dialog for registering the Verifier Software

3. To register, the user has to enter a Registration Code as shown below.
Send the Registration ID to your Philips support contact address.
The support personnel distribute this Registration Code, matching the Registration ID shown in the Registration dialog:

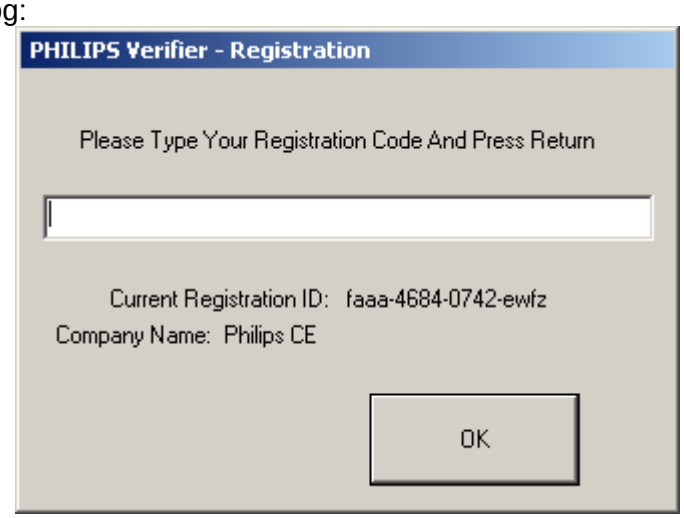


Figure 3-5: Dialog for entering registration code

4. The code entered is validated.
   If found invalid, the user gets the message shown in figure below:

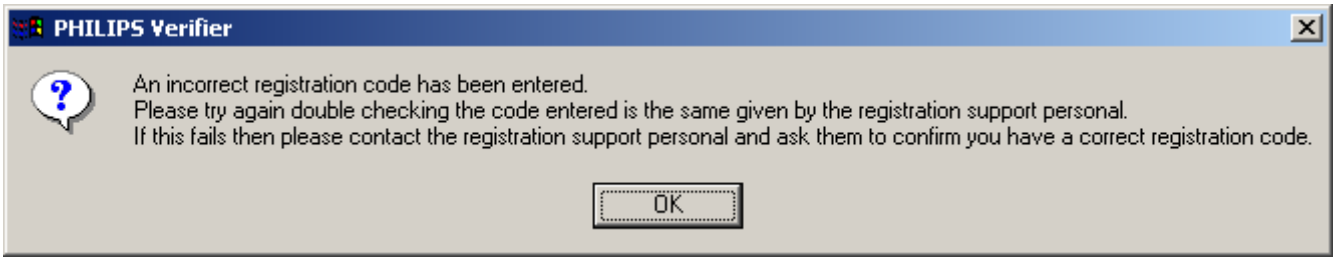


Figure 3-6: Message for an invalid Registration Code

But if the code entered is valid, this message is given:

Figure 3-7: Successfully registered

5. If the user selects the 'Register Later' option, the following message is shown:

Figure 3-8: Message on 'Register later'

But if the user chooses not to register, the verifier software will have to exit.

3.3 DESCRIPTION OF DVD+RW VERIFIER

3.3.1 Open Disc Button

When this button is clicked, the following dialog box is presented allowing selecting the DVD-ROM drive containing the DVD+RW Video disc to verify:

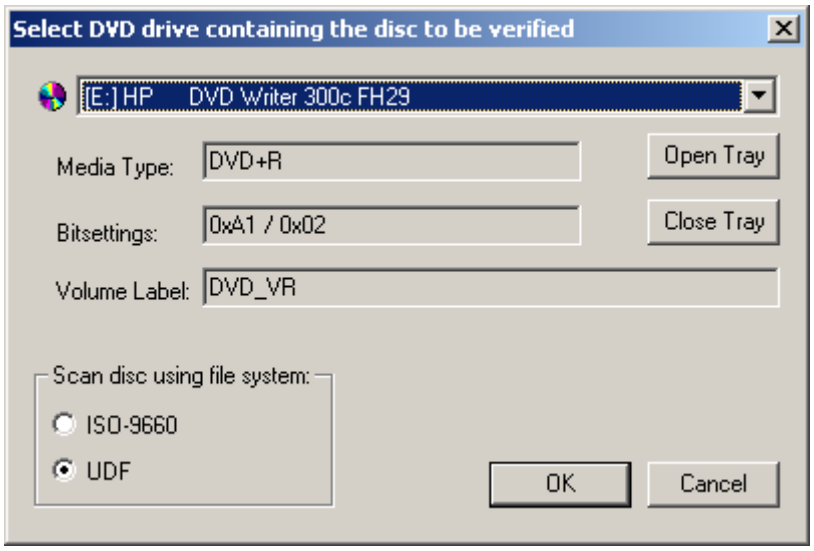


Figure 3-9: Select Drive Dialog

This dialog allows the selection of CD/DVD devices only. Select the DVD drive you want to use for the verification. It is preferred to use a DVD+RW burner device.

The 'Media Type' field indicates the type of media inserted (DVD-ROM, DVD-R, DVD+R, DVD-RW, DVD+RW, DVD-RAM).
In the 'Bitsettings' field, you can see the exact bitsettings for the current disc.
The hexadecimal numbers represent: Booktype & Part Version (0x92) and the Disc Structure (0x04).

Here is an overview of bit settings:

Type of Disc	**Book Type and Part Version (1st byte of Control Data) (hex)**	**Disc structure (3rd byte of Control Data) (hex)**
DVD-ROM single layer	0x01	0x01
DVD-ROM dual layer PTP	0x01	0x21
DVD-ROM dual layer OTP	0x01	0x31
DVD-RAM 2.6	0x11 (version 1.0)	0x04
DVD-RAM 4.7	0x15 (version 2.0)	0x04
DVD-R 3.9	0x21	0x02
DVD-R 4.7	0x25	0x02
DVD-RW	0x31 (v1.0) or 0x32 (v1.1)	0x02
DVD+R	0xA1 (or 0x01)	0x02
DVD+R dual layer	0xE0 (or 0x01)	0x32
DVD+RW	0x92	0x04

On selecting the correct DVD-drive, which contains the DVD(+RW video) disc to be verified, the application scans the disc to detect all DVD+RW compliant files it contains. These are then listed in the main window file list box, transcripted to reflect their contents (cf. 3.3.3 Files).

Remark 1: Be carefull to select a DVD drive only.
CD drives normally cannot read DVD discs.

Remark 2: Some DVD-ROM drives cannot read DVD+RW or DVD-RW discs.
Please check the specifications of the DVD-ROM drive.

Remark 3: A DVD+RW drive is prefered to be used as verification drive, because legacy DVD-ROM drives may not properly read DVD+RW discs.

3.3.2 Open File(s) Button

When this button is clicked, the following dialog is presented allowing the user to select the location of the files to be verified.

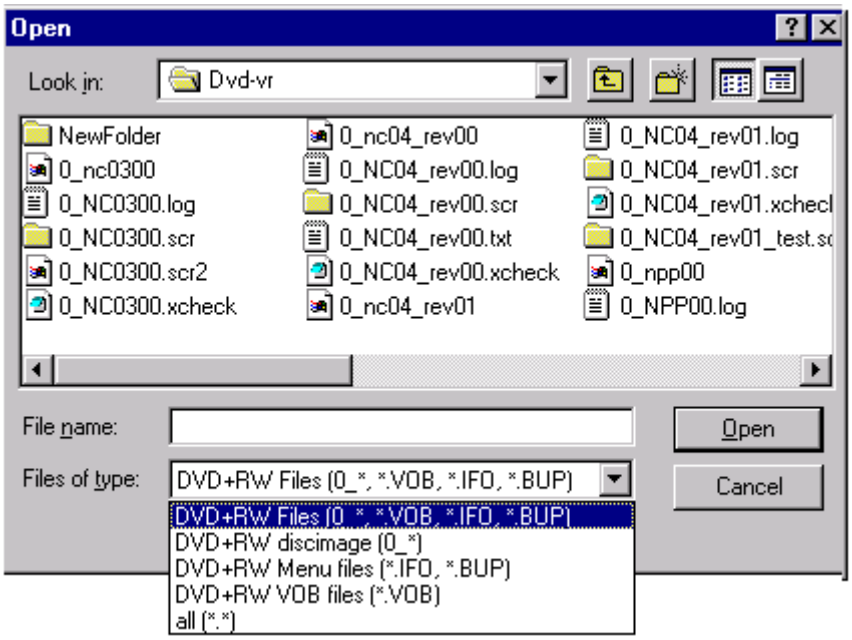


Figure 3-10: Open File(s) Dialog

The 'Files of type' allows the user to select which set of files should be showed in the file listbox. To change the selected file type, click on the arrow button and select a file type from the dropdown listbox.

Figure 3-11: Files of type

The following types of DVD+RW files are supported.

- 'DVD+RW Files' includes all DVD+RW specific files, which are supported. '0_*' for disc images, '*.VOB' for VOBS files, '*.IFO' for Menu files and '*.BUP' for Backup files of the Menu files.
- 'DVD+RW discimage' includes only the disc image files conforming to the '0_*'-prefix or '1_*'-prefix naming convention. For verification, only the '0_*'-file must be selected, the corresponding '1_*'-file will be verified automatically.
- 'DVD+RW Menu files' includes only the DVD+RW Menu files, '*.IFO' for Menu files and '*.BUP' for Backup files of the Menu files.
- 'DVD+RW VOB files' includes only the DVD+RW VOBS files.
- 'All' includes all files.

The 'Files' list box in the main DVD+RW dialog window lists all candidates found in the currently selected directory.

Select the disc image or the DVD+RW files (multiple files can be selected using CTRL and SHIFT), which should be included in a verifier run.

When a disc image is selected, the program scans it to detect all files it contains. These are then listed in the main window 'Files' listbox, transcripted to reflect their contents (cf. 3.3.3 Files).

When scanning of the disc image takes a long time, a window is shown temporarily, to indicate that the verifier is busy.

When a file is selected with a filename similar to a disc image (0_*, 1_*) name, and during scanning this file it was determined that it was actually no disc image, the file will also be added to the file list but will be verified as a PS!

When a set of DVD+RW files has been selected, these are also copied to the parent window file list window and transcripted.

When a disc image is selected together with other files, all other selected files will be discarded and only the disc image will be scanned and verified.

3.3.3 Files

Lists all files that can be verified, e.g.:

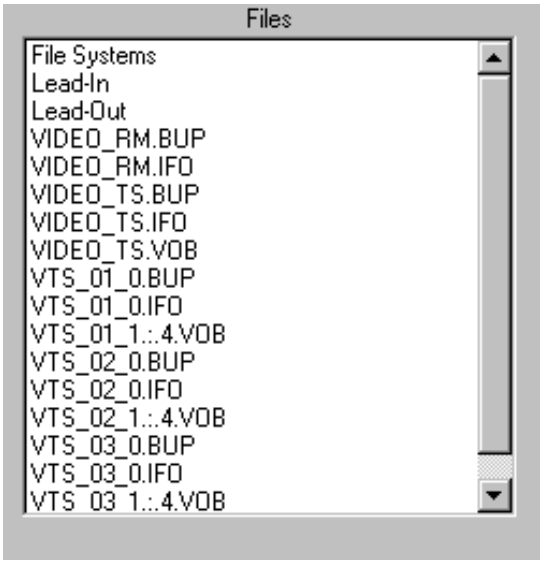


Figure 3-12: Main window file list box with transcripted disc contents

This allows you to select the file(s) to verify.

Files belonging to a VOBS are represented in a special way: VTS_***xx_y***.:.***z***.VOB, where 'xx' equals the VTS number, 'y' equals the first VOBS file (normally '1') and 'z' equals the last VOBS file. This is the so-called file name "**transcription**". When this entry is selected, all files belonging to the VTSTT_VOBS will be verified.

3.3.4 Select All

Selects all possible candidates listed in the 'Files' for verification.

3.3.5 Deselect All

Deselects all files listed in ‘Files’ for verification.

3.3.6 Start

Starts verification of all selected files with the chosen options. If no files have been selected, a requester will inform the user of this problem.

If the settings are not changed before the verification is started, the settings of the previous run will be used.

When clicked, the caption on the button will change to ‘Stop’, and some buttons on the main window will be disabled in order to prevent the user from selecting them.

The [Stop] button can be used to abort the verification. When clicked, the following message appears:

Figure 3-13: Quit Dialog

Verification will continue even while this requester is displayed, until the user selects the [Yes] button.

3.3.7 View Log Files

When the verifier has finished the complete verification process, the view log files window (Lumberjack) is presented. This window shows the output log files generated by the verifier. The user can select a specific log file on the left hand side of the window, while the log file text is displayed on the right hand. It is possible to view and scroll through the log file text.
Per log file a summary of the messages (informations, warnings, errors, etc.) is to be found at the end of each log file.

The path of the selected log file is made visible on the title bar.

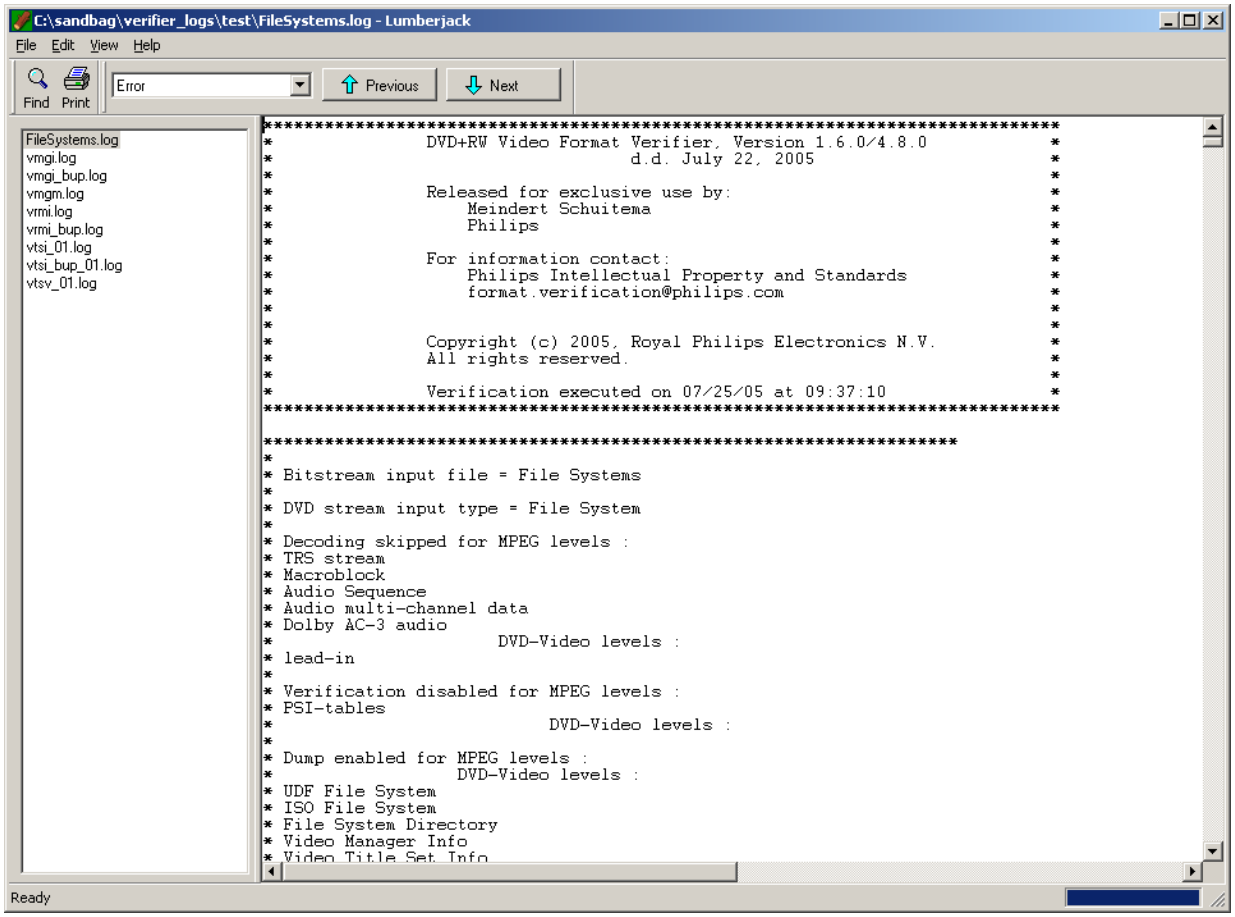


Figure 3-14: View Log Files

When the view log file window is closed, it can be re-opened by clicking on the button [View logfile(s)] on the main screen of the verifier.

Within the view log file window, you can search on a specific message type with the buttons Next and Previous.

It is also possible to search for a specific piece of text with the Find button.

Each log file can be printed.

It is possible to view very large files (up to 5 Gbytes).

3.3.8 Show Error Report

When clicked, this button enables the window displaying a list of all messages generated since the start button was clicked, each with its type and the number of times it has been reported so far. The errors are sorted per type and number. This window is updated continuously.

The caption of the button changes to “Hide Error Report”, which when clicked, closes the error report window.

The error count information is only reset at the click of the [Start] button, and is not changed by successive Hide and Show clicks. So at the end of a verifier run it will contain the cumulative error count over all verified files.

Note that the error report window can only be closed by clicking the [Hide Error Report] button.

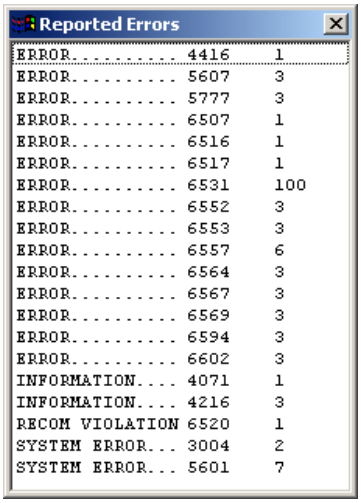


Figure 3-15: Error report window

3.3.9 Show Output

When clicked, this button enables the real-time output window, so the output being generated and stored in the log files can be watched on screen too.

The caption of the button changes to “Hide Output”, which when clicked, closes the output window.

Remark: The output log information is lost when the [Hide Output] button is clicked. When again clicking the [Show Output] button an empty screen is presented.

Note that the output window can only be closed by clicking the [Hide Output] button.

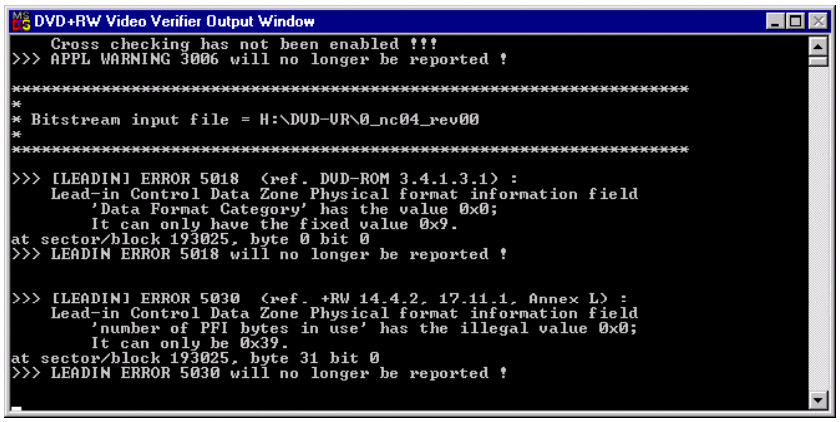


Figure 3-16: The output window

3.3.10 Progress bars

At the bottom of the main window are two progress bars. The top progress bar shows the progress of the file currently being verified. The caption above this progress bar indicates the filename of the file (“…” when the program is idle), which is currently being verified.

The bottom progress bar shows the progress of the verification of all selected files. When the bottom progress bar reaches 100%, verification has finished.

3.3.11 Status Bar

The Status Bar is located in the lower left corner of the window:

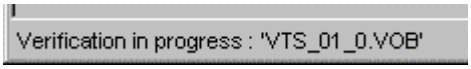


Figure 3-17: The status bar

The Status Bar is used to display what the DVD+RW verifier is doing.

3.3.12 Verification Termination

If for some reason the verifier encounters an insurmountable problem, the program will quit verification and displays the following message:

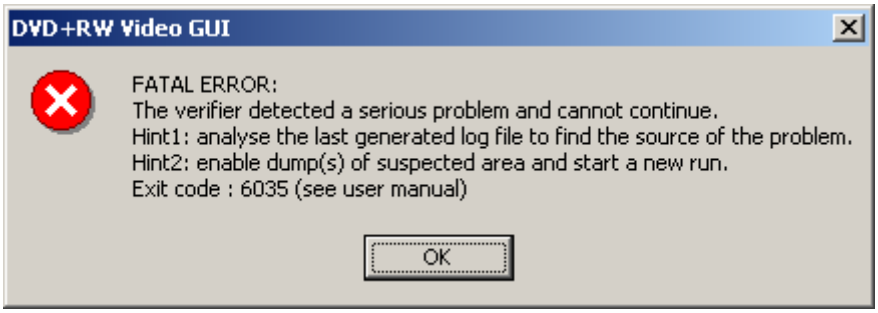


Figure 3-18 : Fatal Error

The exit code displayed in the requester indicates the nature of the problem.
A description of all exit codes can be found in Chapter **8.4**. **Exit codes**.

3.4 THE 'MENU'

3.4.1 The File menu

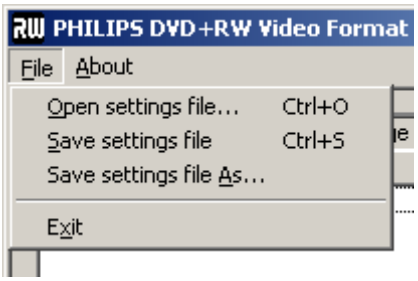


Figure 3-19: File menu

Open settings file…	Open a Settings file. A standard windows file requester window appears, which can be used to select the settings file to be read.
Save settings file	Saves the current settings to the filename displayed in the title bar of the main window. If no filename was previously selected, a standard Windows file requester will allow the user to select the filename.
Save settings file As…	Saves the current settings to a selectable filename. A standard Windows file requester will allow the user to select the filename.
Exit	Exits the program (or click on the button in the right top corner of the window).

3.4.2 The Help menu

Figure 3-20: Help menu

About DVD+RW_GUI… Shows version information about the DVD+RW verification tool.

When clicked, this menu will show the current verifier's version number and copyright notices:

Figure 3-21: About box

3.5 DETAILED DESCRIPTION OF VERIFIER SETTINGS

The 'Settings' window

The settings window is shown when the Settings button in the Toolbar is clicked, or the Settings from the View menu is selected.

The Settings window is subdivided into 6 tab windows. These tab windows group settings for easy reference. Initially, the Miscellaneous settings tab is displayed.

Upon clicking the [OK] button, the settings are saved in the system's registry, where these are again retrieved from at the next execution of the verifier.

3.5.1 Misc settings

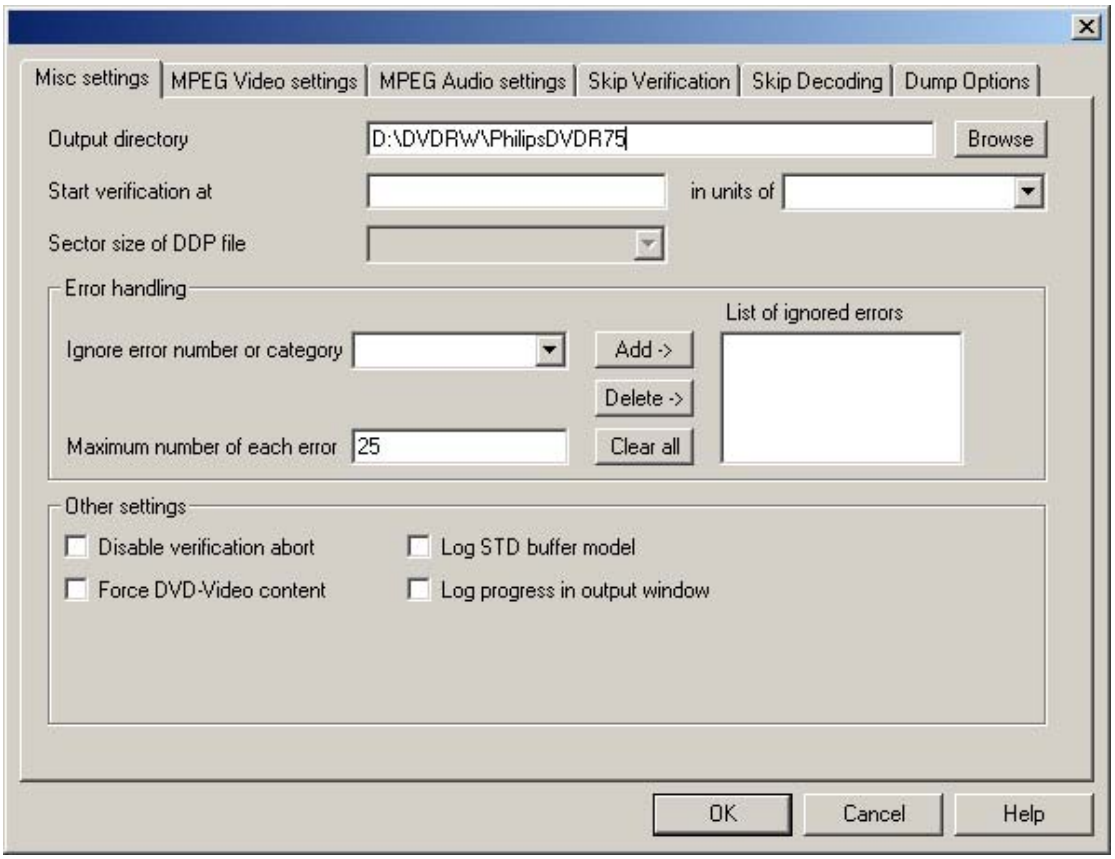


Figure 3-22: Miscellaneous Settings

3.5.1.1 Output Directory

This option is used to specify the output directory of all generated files: contents dump, verification report log files and STD buffer log files. If no output directory is specified, the files are written to the same directory as where the input files reside. However this is obviously not allowed when verifying directly from DVD+RW disc, or when the input files reside in a read-only directory. In this case one has to specify some directory with write access.

The user can directly type the output directory in the editbox, or can click the Browse button, which will present a standard Windows directory requester.

When the [Start] button on the main windows is clicked, the program checks if the selected output directory is writable and informs the user with a requester if this is not the case.

Remark: The specified output directory must be terminated by a slash.

3.5.1.2 Start verification at

This option allows starting the file parsing & verification process not immediately at the start of a file but only from a specified location onwards, e.g. to speed up the, in some cases lengthy, VOBS verification. Specify in this field the access unit number or stream position to start the verification at.

This option is only intended to be used for (title) VOBS data and is ignored for other, navigation data files.

Note: In case of disc(image) input, the specified start location will be used for ALL VOBS files on the disc(image). It is not possible to specify a different location for each VOBS.

➔ **Beware**: This option should be used with extreme care and only to analyse really problematic data streams! It may cause some invaluable or even required data to be skipped and thus missing for the verification process. This may at least result in some unexpected (and unjustified) error messages. Also variable length coded data (e.g. MPEG or Dolby AC-3 audio) may well be out of sync, causing parser syntax errors and in an exceptional case a program crash. It is strongly recommended to specify the start position of a VOBU and preferably a Cell, because then side effects are limited to the minimum. Anyhow some checks will be disabled (cf.). Needless to say that the stream's buffer verification is no longer valid.

3.5.1.3 in units of

This specifies the units used by the start position above. This can be:

- "sector (PSN)" in case of disc(image) input
- "pack (RLBN)" in case of a VOBS file (which is in fact an MPEG Program Stream).
- "packet" in case of PES (Packetized Elementary Stream) input.
- "byte" in case of ES (Elementary Stream) input.

3.5.1.4 Sector size of DDP file

In case of a DDP disc image input of which the matching ID file is not available or corrupted, the verifier tries to determine the sector size used by the image (since several variants exist). If this does not succeed, e.g. when there are several possibilities, the verifier reports this and the user is requested to specify the correct sector size through this field. The user can select a value from the dropdown listbox.

This is currently not yet implemented.

3.5.1.5 Ignore error number or category

The user can either force the verifier to ignore all messages of a certain class (informations, errors, syntax errors, system errors, warnings, oddities, and recommendation violations) or ignore a specific error by typing its error number. Click the 

button to transfer the selected error class or error number to the 'List of ignored errors' listbox.

3.5.1.6 List of ignored errors

This listbox lists all currently selected errors to be ignored during the verification process. To remove an error class or error number from this listbox, select the entry and click the 

button. To remove all entries from this listbox, click the 

button.

3.5.1.7 Maximum number of each error

Specifies the maximum number of times a specific error message should be logged. This is set to 1 by default.

Note however that this has no impact on the final error report, where the total number of occurrences of each error is reported.

3.5.1.8 Disable verification abort

By default, the verifier will abort the verification process when a specific combination of errors has occurred which most probably indicates the presence of "bad spots" and related write and/or read errors (cf. 12 Defective Media Handling). This would typically cause a "waterfall" of

even fundamental errors and hence a huge amount of error messages. To avoid this premature exit, this button can be checked.

3.5.1.9 Log STD buffer model

When this button is checked, the program creates an ASCII file for every PES (characterized by a specific stream_id) in a VOBS file, to store the STD-buffer model verification data. private_stream_1 or private_stream_2 PES streams are further differentiated by their sub_stream_id and assigned a file per sub_stream_id.

The STD-buffer log files use the following naming convention:

File type	File name
PES streams	<filename>.std_<stream_id>
Private streams	<filename>.std_s<sub_stream_id>

These files can be fed to the public domain tool “gnuplot”, or can be imported into Microsoft Excel, to generate a graphical representation of the P-STD buffer contents of a particular PES over the complete input stream.

To import one of these files into Microsoft Excel, select ‘File->open’ from the menu. Select the filename of the STD-buffer you want to load. The text import wizard will pop up, in which you can use all the default options. You can then insert a chart, which uses the A1 column as the X-axis and the B1 column as the Y-axis. Use the menu ‘Insert->Chart’ and choose the ‘XY scatter’ or ‘Line’ type. For some reason the data range is limited to 32000 samples per series.

3.5.1.10 Force DVD-Video content

It is possible a DVD+R, DVD-R or DVD+RW, DVD-RW disc contains DVD-Video content. If such disc is verified with this verifier, a lot of errors will be generated because the disc will not be compliant with the DVD+RW Video Format specification.
With this verifier it is still possible to verify a DVD-Video disc or a disc with DVD-Video content. Be sure to enable the ‘Force DVD-Video content’ selection.

3.5.1.11 Log progress

This option, when checked, adds a line to the output log file about every 100 access units (sectors, packs or packets) to give an idea of the verification process progress. In a way it duplicates the function of the progress bars.

3.5.2 MPEG Video settings

These settings allow to enter “a-priori” knowledge of the disc or file contents.
This enables some additional checks, e.g. cross checks with the actual DVD+RW AV-data.
Use these settings only when verifying a single .VOB file or a set of .VOB files.

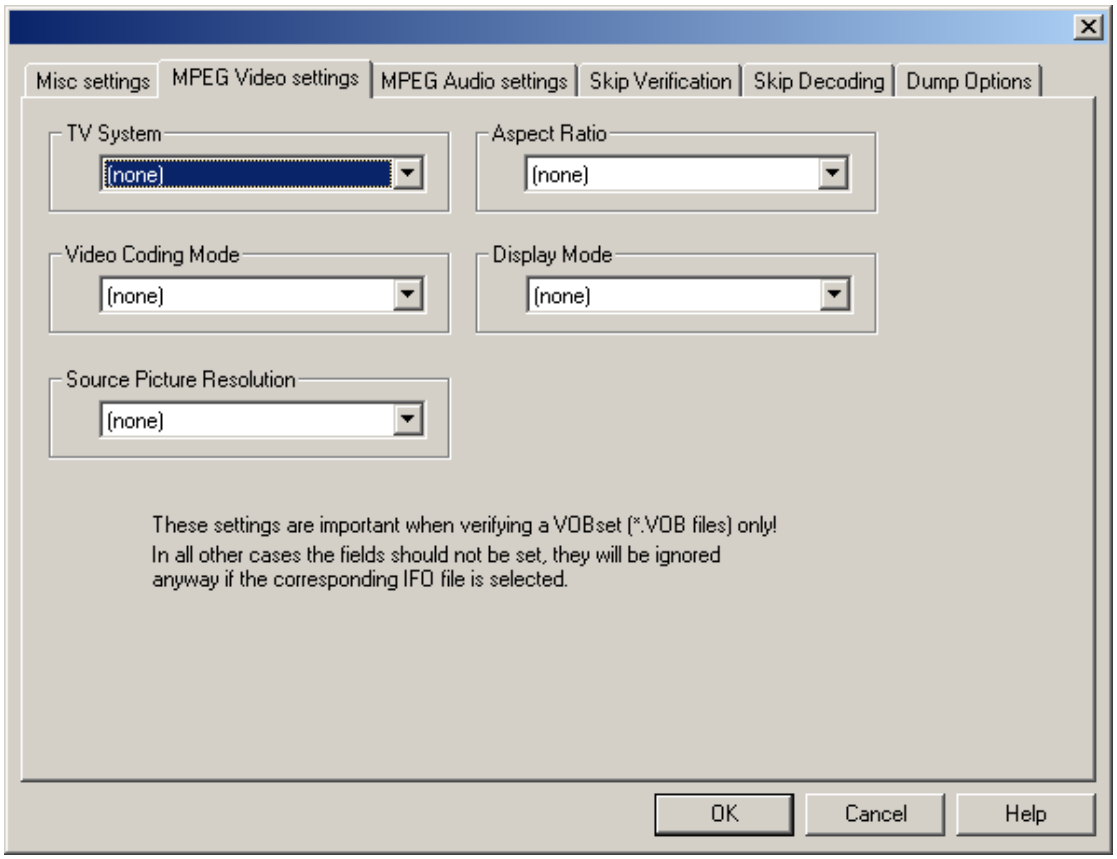


Figure 3-23: MPEG Video settings

3.5.2.1 TV System

This list box allows the selection of PAL or NTSC (or none).

When there is no VMGI or VTSI derived cross check data available, explicitly specifying a TV system really is necessary. Without proper cross check data and if the user has no TV system selected (selected ‘none’), the verifier lacks the correct information about the TV system in use on the disc, and some checks will not be performed and some unjustified error messages may be reported.

By default NTSC is assumed.

3.5.2.2 Display mode

This list box allows selecting:

- None
- Both pan-scan and letter box
- Only pan-scan
- Only letter box.

3.5.2.3 Aspect ratio

This optional setting allows selecting either 4:3 or 16:9 aspect ratio (or none).

3.5.2.4 Source picture resolution

This optional setting allows selecting various video picture resolutions:

- None
- 720 x 480 (ntsc), 720 x 576 (pal)
- 704 x 480 (ntsc), 704 x 576 (pal)
- 352 x 480 (ntsc), 352 x 576 (pal)
- 352 x 240 (ntsc), 352 x 288 (pal).

By default 720 x 480 is assumed.

3.5.2.5 Video coding mode

This optional setting allows to select either MPEG-1 or MPEG-2 video encoding (or none).
By default MPEG-2 video is assumed.

3.5.3 MPEG Audio settings

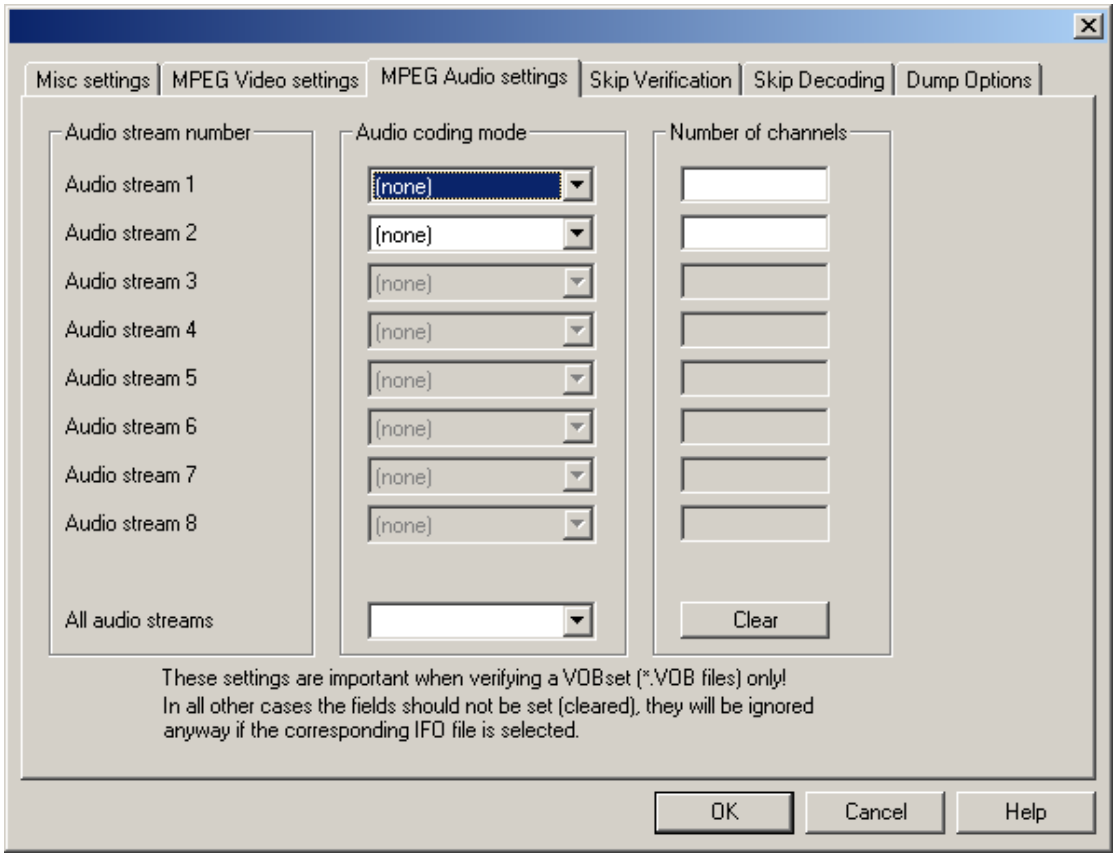


Figure 3-24: MPEG Audio settings

The Audio settings window allows specifying the MPEG audio encoding type (MPEG-1 or MPEG-2) and the number of audio channels for each MPEG encoded audio stream present in the selected VOBS file.

This is only necessary in case there is no matching VMGI file (for the VMG menu VOBS VIDEO_TS.VOB) or VTSI file (for VTS menu VOBS or a title VOBS), where this information can be retrieved from.

The ‘All audio streams’ entry can be used to select the audio type and number of channels for all audio streams at once. The values for all audio streams will change immediately, allowing overriding the settings if necessary.

Since DVD+RW Video discs only allow for two audio streams, the entries for audio streams 3..8 are disabled.

Remark: The LFE channel, commonly noted as “.1” is counted as 1 additional channel, so “5.1” must be entered as 6 channels.

3.5.4 Skip Verification

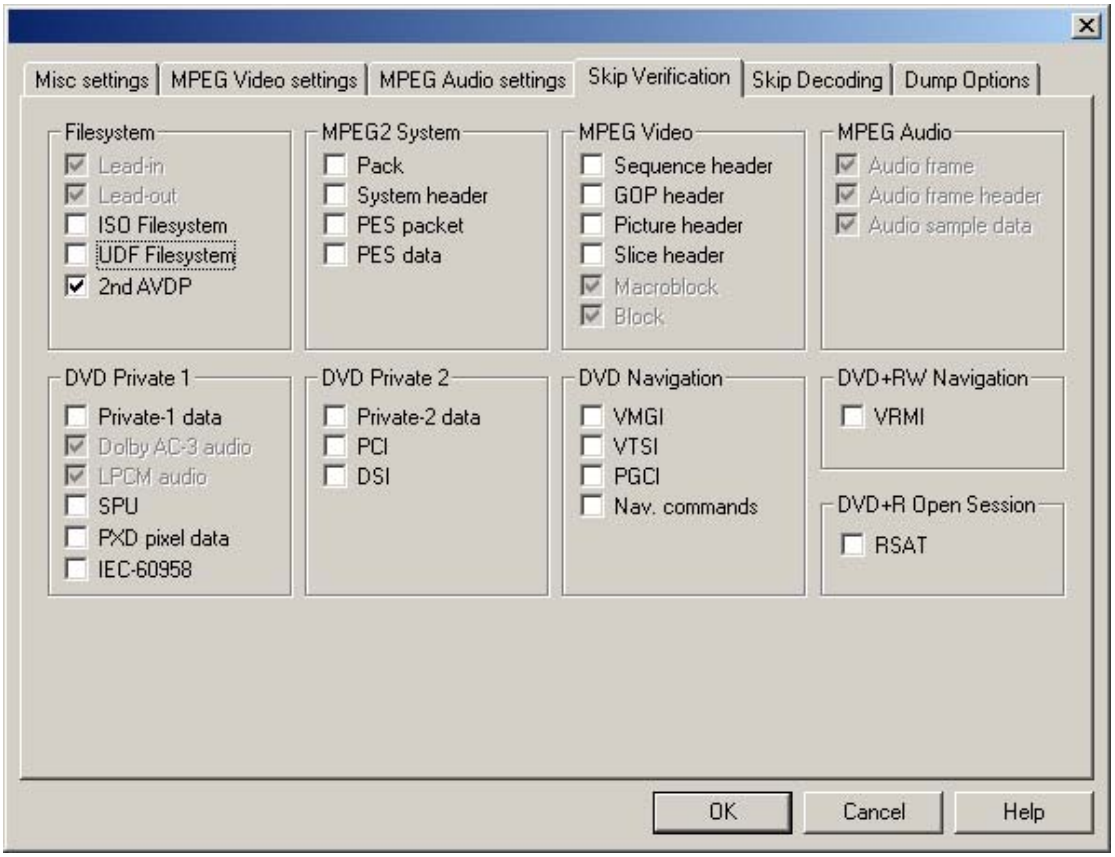


Figure 3-25: Skip Verification settings

The skip verification settings determines the levels that will not be verified.

This can be used to ignore errors from a particular level of MPEG and/or DVD+RW data. Levels that are not parsed (selected in the 'Skip decoding' settings window) appear ghosted and are automatically skipped for verification too.

Checking the '2nd AVDP' button allows to skip parsing and verification of the UDF file system's 2nd "anchor point" (AVPD). This may be useful when verifying an actual DVD+RW disc: The 2nd AVDP must be recorded at the very last written sector of the disc's Data Zone. However, it may be incorrectly located or even missing which then drives the verifier into a disc's "ice" when trying to read the AVPD sector, causing many retries or even a drive hang-up. Through this option this problem can be avoided.

➔ **Beware**: The default settings are as shown in figure 3-25. If other settings are used, this may lead to less or more checking and it is possible the verifier does fully verify the disc. So be aware of this!

3.5.5 Skip Decoding

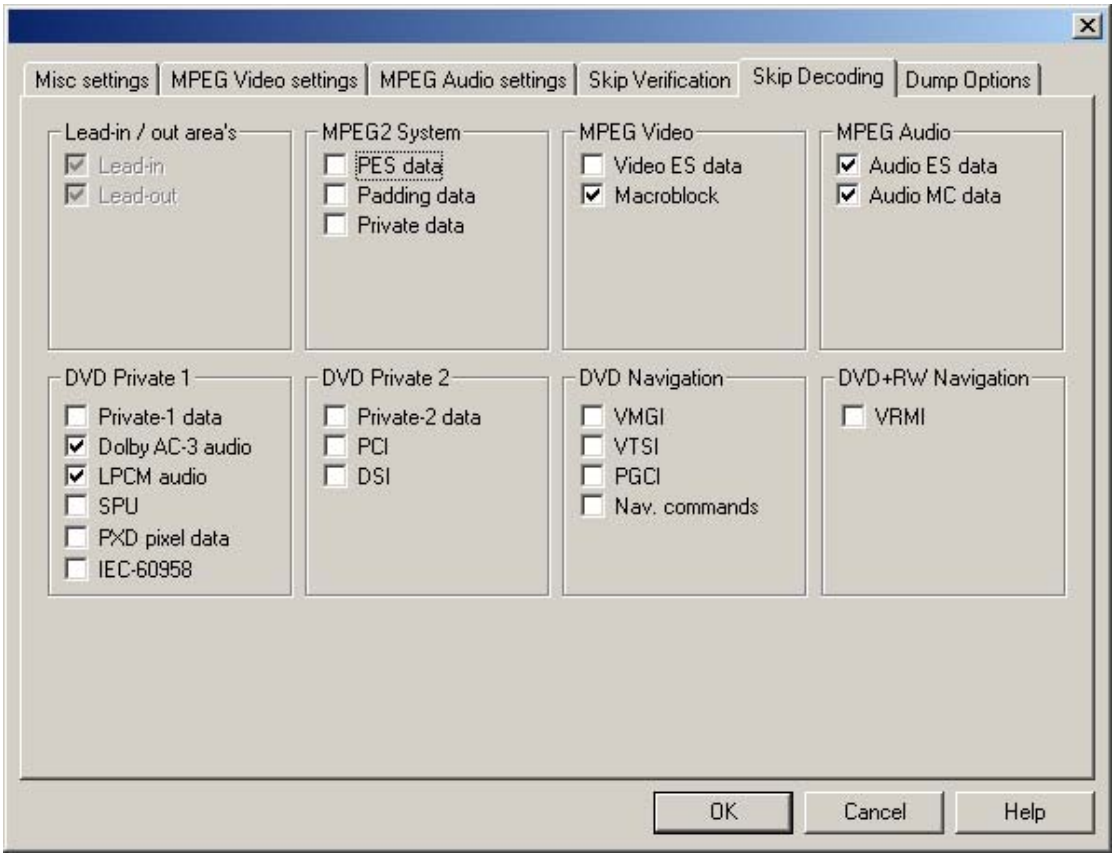


Figure 3-26: Skip decoding settings

The 'Skip decoding' settings window allows specifying the data levels that should not be parsed.

Sometimes it is useful not to decode the complete data stream, especially because not parsing some data elements can have a dramatic impact on the speed of the verification. Once a level is being skipped here, it cannot be selected for contents dump in the Dump settings window and verification of this level will automatically be skipped too. Also, sub levels of the level selected in this window are affected. All affected levels are grayed-out in the Skip verification and Dump settings windows.

➔ **Hint**: Selecting the MPEG Video Macroblock data level speeds up the verification process significantly. Also skipping Dolby AC-3 audio and LPCM audio parsing significantly speeds up verification.

3.5.6 Dump options

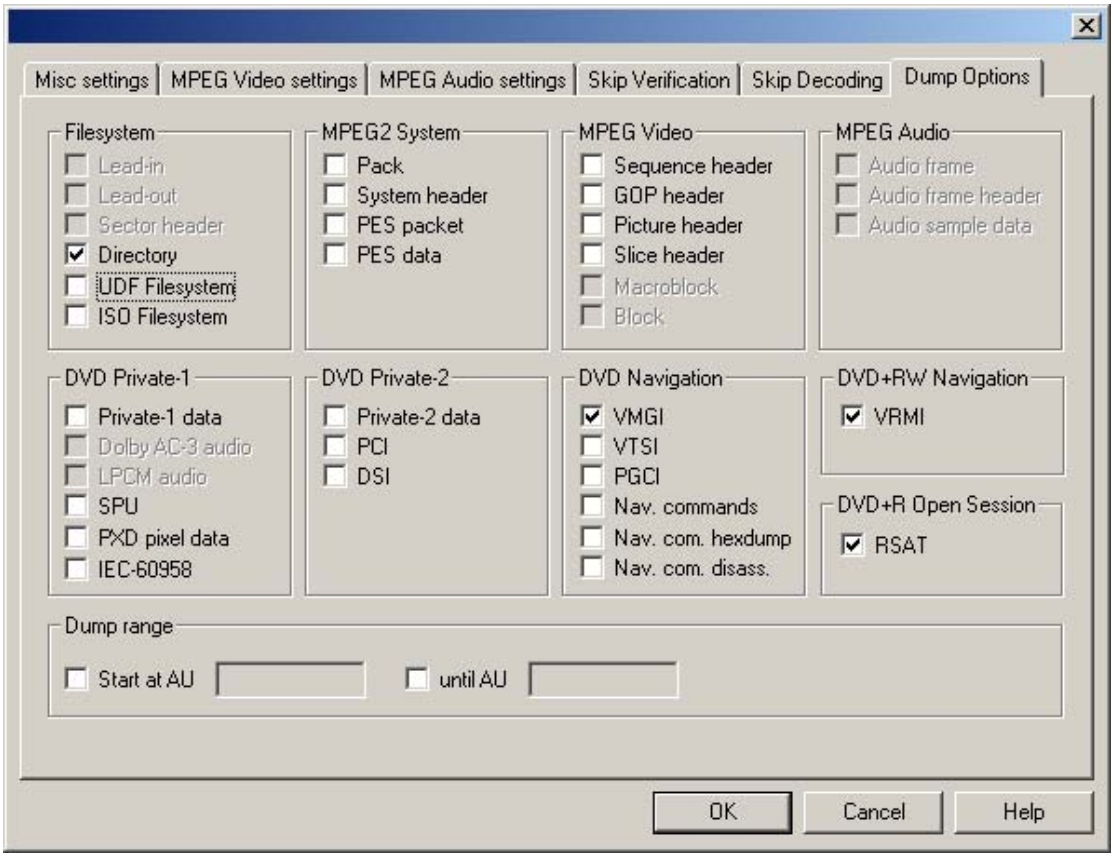


Figure 3-27: Skip decoding settings

The Dump options window allows specifying the data levels to generate a contents dump (bit stream disassembly) for. Levels that are not parsed (selected in the 'Skip decoding' settings window) appear ghosted and are not selectable.

The Dump range settings (cf. below) are used for limiting the data generated when a dump option is selected. If selected, enter the number of the first access unit to be dumped in the first editbox and enter the number of the last access unit to be dumped in the second editbox.

It is not necessary to specify both start and end ranges.

3.6 AUTOMATED VERIFIER RUNS

In order to facilitate automatic verifier runs from a batch (script) file, the program can be used to run automatically with pre-set settings. Whenever the program is started with the command line option '**-S**<filename>', the program will start and end without the need for user intervention (the window still will show and the user still can select the error/output window). The '<filename>' must be the name of a valid settings file which was previously saved (Menu:File->Save).

To prepare for an automated verifier run, first start the DVD+RW GUI normally and set all necessary settings. When all settings are set, save them to a file (*.vrset). Alternatively, test if the settings are correct by running the verification and then save the settings to a file. Use the filename of this file in the '-S<filename>' command line option.

3.7 COMMAND LINE INTERFACE

3.7.1 Synopsis

dvdvr_verf [-**G**(vaPEVA12rmtgu?ypdDxFM)] [-**Y***driveletter*] [-**W**]
[-**X**(ARMTVFLOf)] [-**V***N*] [-**e***N*] [-**x**(IORWXES)] [-**p**]
[-**s**(KYLXVGPSMBFHAs0mtg1uxay2pdcfir^)]
[-**d**(KYLXVGPSMBFHAs0mtg1uxay2pdcnhfizr9)]
[-**S**(PEVMACUs0mtg1uxay2pdcrfi)]
[-**A**] [-**C**] [-**M**(RONE)] [-**P**(CV)] [-**U**(IB)] [-**g**] [-**a**(127)] [-**m**(12)] [-**r**(0123)] [-**z**(01)]
[-**B**] [-**b**] [-**h**] [-**v**] [-**f***N*] [-**n***N*] [-**N***N*] [-**o***N*] [-**O***N*]
[-**D***scriptfile*] [-**E***extfile*] [-**L***logfile*] [-**Z***outputdir*] *inp_filename*

It is recommended to specify all options in a script file, which can be fed to the verifier with the –**D***scriptfile* option. If an option is specified on both the command line and in a script file, then the latest (scanning from left to right, top to bottom) specified options take precedence.

3.7.2 Command Line Options

When combining the various command line options, the following rules must be considered:

- Options may appear in any order; they may even follow the input file name.
- There may be no space between an option and its parameter.
- Options and their parameters are case sensitive!
- If an option does not take a parameter, it may immediately be followed by another option.
- When incorrect option or parameters or no command line options at all are specified, the verifier simply echoes all allowed options, as some kind of (minimal) help. Somewhat more extensive help is obtained using the "-**h**" option.

Option	Para meter	Description	remark
-**G***type*		**Input stream type**: (default: **P**)	1
	v	MPEG-1 video	
	a	MPEG-1 audio	
	P	MPEG-2 PS	
	E	MPEG-2 PES	
	V	MPEG-2 video	
	A	MPEG-2 audio	
	1	MPEG-2 private_stream_1	
	2	MPEG-2 private_stream_2	
	r	VRMI	
	m	VMGI	
	t	VTSI	
	g	PGCI	
	u	SPU	
	?	Dolby AC-3	
	y	Linear PCM	
	p	PCI	
	d	DSI	2, 3
	D	DVD+RW disc or (DDP/CMF) disc image	
	F	(UDF + ISO) File System(s) "on file"	4
	M	BEE cartridge	5
-**Y***driveletter*		Read from the DVD-ROM drive specified by *driveletter*	6

-**W**		Specify use of s special verification ROM drive, in combination with the –**Y** option	7, 8
-**X***subset*		**Specify data subset**: (default**: All**)	
	A	All data on a disc(image), incl. Lead-in and Lead-out	9
	R	VRMI	
	M	VMGI and VMGM_VOBS	
	T	as “-M” + VTSI and VTSM_VOBS	
	V	as “-T” + Title VOBS	
	F	specified files only	10, 11
	L	Lead-in	10, 12
	O	Lead-out	10, 12, 13
	f	File system(s) data only	
-**V***N*		Specify the VTS number to process (default**: All**)	14
-**e***N*		Report max N messages per error (default**: 1**)	15
-**x***type*		**Ignore messages of specified type**: (default: **none**)	
	I	ignore Info	
	O	ignore Oddities	
	R	ignore Recommendations	
	W	ignore Warnings	
	X	ignore Syntax Errors	
	E	ignore Errors	
	S	ignore System Errors	
-**p**		Show progress info	16
-**s***level*		**Skip verification of the specified levels**: (default: **none**)	
		MPEG levels:	
	K	pack	
	Y	system_header	
	L	PES_packet	
	X	PES_packet data	
	V	sequence_header	
	G	GOP_header	
	P	picture	
	S	slice	
	M	macroblock	
	B	block	
	F	audio frame	
	H	audio frame header	
	A	audio data	
		DVD levels:	
	s	Sector	12
	0	Lead-in & Lead-out	12, 13, 17
	m	VMGI	
	t	VTSI	
	g	PGCI	
	1	private_stream_1 data	
	u	SPU	
	x	PXD data	
	a	Dolby AC-3	
	y	Linear PCM	
	2	private_stream_2 data	
	p	PCI	
	d	DSI	
	c	Navigation Command	
	f	ECMA/UDF file system data	10, 18

	i	ISO-9660 file system data	10, 18
		DVD+RW Video levels:	
	r	VRMI	
		Special test cases:	
	^	UDF 2nd anchor point (AVPD)	19
-d*level*	as **-s** level plus	**Generate dump of the specified levels**: (default**: none**)	
	n	Disassembled Navigation Commands	
	h	Hex dumped Navigation Commands	
	z	Generate file system derived directory listing	10, 18
	b	bit settings (cf.6.1 dump Bit Settings)	8, 20
		Special test cases: (cf. 6.3 Sector Scan)	
	91	Scan non-zero sectors and hex dump contents	21
	92	Scan sectors and hex dump their contents	21
-**S***level*		**Skip decoding of the specified levels**: (default**: none**)	22
		MPEG levels:	
	P	PS data	
	E	PES data	
	V	Video ES data	
	M	Video ES macroblock data	
	A	Audio ES data	
	C	Audio ES multi-channel data	
	U	User private data	
	as **-s** level plus	DVD(+RW) levels:	
	f	ECMA/UDF file system data	10
	i	ISO-9660 file system data	10
-**A**		Skip verification of DVD(+RW)Video specific constraints	23
-**C**		Disable cross-checking	24
-**M***type*		**Specify target medium type** (default**: R**)	25
	R	DVD+RW disc with DVD+VR content	26
	O	DVD-ROM disc with DVD+VR content	27
	N	DVD+RW disc with DVD-Video content	28
	E	BEE cartridge	29
-**P***type*		**Specify parser control method** (default**: C**)	45
	C	(VTSI) Cell controlled	
	V	(VRMI) VR Play List controlled	
-**U***type*		**Specify IFO or BUP data to use** (default**: IFO**)	46
	I	Original (IFO)	
	B	Backup (BUP)	
-**g**		Overrule premature program abort	47
-**a***N*		Interpret all MPEG audio as (default**: MPEG-1**) **1**:MPEG-1, **2**:MPEG-2, **7**:Augmented	30, 31
-**m***N*		Interpret all Video as (default**: MPEG-2**) **1**:MPEG-1, **2**:MPEG-2	30
-**r***N*		Set picture resolution (for all video data) as (default**: 0**) **0**: 720x480or576, **1**: 704x480or576, **2**: 352x480or576, **3**: 352x240or288	30
-**z***N*		Specify TV-system as (default**: NTSC**) **0**: NTSC / **2**: PAL	30
-**B**		Generate plot files of P-STD Buffer model	32
-**b**		Generate plot file of VBV Buffer model	32, 33
-**h**		Show some help information	34

-**v**		show program header (version information)	35
-**f***N*		Start parsing at LSN or RLBN (*N*)	37,39,36,40
-**n***N*		Start contents dump from LSN or RLBN (*N*)	37,39,41,40
-**N***N*		Stop contents dump at LSN or RLBN (*N*)	37,39,41,40
-**o***N*		Skip parsing first *N*-1 bytes	36, 38
-**O***N*		Stop parsing at *N* bytes	38
-**D***scriptfile*		Use script file *scriptfile* (cf. **3.8 Script File Interface**)	
-**E***extfile*		Use extension bit stream *extfile*	42
-**L***logfile*		Copy screen output to log file *logfile*	43
-**Z***outputdir*		Full path (ending with slash) specifying output directory.	
inp_filename		Use data input file *filename*	44

Remarks

All command-line options may be combined with the use of a script file (cf. **3.8 Script File Interface**).

As a general rule, in case of conflicts between user command-line settings, later specified (scanning from left to right) settings may overrule any preceding settings.

0. This option is not yet implemented.
1. All input files with (stream) data to be verified are simple binary files.
2. When no (DVD-ROM or verification) drive is specified (e.g. using the –**Y***drive* option), a DDP/CMF disc image input is assumed.
3. In case of DDP/CMF disc image input, the specified input (disc image) file name *filename* must be conform the old DDP file naming conventions (i.e. one of “0id_<name>.toc”, “0lc_<name>.toc” or “0_<name>”) or conform the CMF file naming convention (i.e. one of “DDVID.DAT”, “CONTROL.DAT” or “IMAGE.DAT”)
4. This option allows to separately parse & verify a disc’s file system data, which is then fed to the verifier on a separate binary file.
5. Basic Engine Emulator (BEE) cartridge is in fact a removable hard disk on which the complete contents of a DVD+RW disc is stored in a specific BEE format (containing only sector payload data).
6. Only relevant in case actual disc input “-**GD**” or BEE image “-**GM**” has been specified.
7. This allows to parse & verify also a disc’s ‘raw’ (physical) data such as the Lead-in, Lead-out and sector header data. Normally it must no longer be specified explicitly since the verifier is now capable to automatically detect the presence of a special verification drive (cf. Appendix B). But this option can still be used to force any drive to be treated as a special verification drive.
8. Only relevant in case actual disc input “-**GD**” has been specified.
9. Lead-in and Lead-out data is only parsed & verified when complete verification or Lead-in resp. Lead-out only verification is specified (resp. “-**XA**”, “-**XL**” or “-**XO**”).
10. Only possible in case of actual disc or disc image input (“-**GD**”).
11. When a subset of some files is specified, the file names have to follow the “-**XF**” options. Furthermore, some disc(image) name *input_filename* has to be specified before the “-**XF**” option, even in case of actual disc input, to avoid the first specified file name interpreted as disc(image) name.

12. Lead-in, Lead-out and sector data parsing & verification are only possible when available, i.e. when reading from a disc image or from a disc using the ASALE verification drive (cf. Appendix B), and irrelevant in other cases. Normally, when not using a special verification drive, Lead-in and Lead-out parsing is automatically disabled.
13. Lead-out is only present on an actual disc or a BEE image, and not on a DDP/CMF disc image
14. Allows to parse & verify only the data part of the specified VTS (Title) number.
15. Checks that are similar for e.g. audio and video data might be reported by the same error number. Typically multiple PES streams are multiplexed into a VOBS. As a consequence these errors may be invisible for one stream because the related error messages are suppressed because the maximum error count has already been reached for another stream of the VOBS
16. Outputs a text line about every 100 top level access units (sector, pack, packet, frame, etc.), to indicate the verification process is still busy and give an idea of its progress.
17. Skipping Lead-in verification also skips reading/parsing the Lead-in “null byte” zones, thus speeding up processing.
18. File system verification and directory list is only enabled when either complete or file system only verification is specified, i.e. “-**XA**” or “-**Xf**”
19. The “-**s^**” option allows to skip parsing and verification of the UDF file system’s 2nd “anchor point” (AVPD). It must be stored at the very last written sector of the Data Zone. However, it may be recorded at an incorrect location or even missing which then drives the verifier into a disc’s “ice” when trying to read the AVPD sector, causing a drive hang-up.
20. This option logs the hex value of the sector header’s first byte and Lead-in PFI bytes 0 and 2, at some critical locations : at the start of the Lead-in (Inner) Disc Identification Zone (PSN 0x2EEC0), Control Data Zone (PSN 0x2F200) and at the start of each file in the Data Zone.
21. This option, originally only a test case, is only available in this command-line version of the verifier.
22. This means:
    for ‘lower’ layers: completely skipping the decoding of the layer. e.g. skip audio stream or padding stream decoding,
    for ‘higher’ layers: skipping as much decoding as possible without interfering the decoding process of lower layers. For example, “skip PES stream decoding” will skip all non ES-stream relevant data and decode only A/V or private_1 streams.
23. The “-**A**” option skips most of the application specific verification of the stream, which may be a pure MPEG stream as well as MPEG data embedded in a DVD(+RW)-Video data format.
24. In case of script file input, enabled by default.
25. Because the same contents can be recorded on different physical media, or the +RW physical medium can be used to store non-DVD+RW data, different sets of checks have to be enabled dependent on the target medium type.
26. +RW disc target : both physical & logical data are checked for compliance with the +RW specifications
27. DVD-ROM disc target : DVD+RW Video logical data on a (replicated) DVD-ROM disc. Physical & logical data are checked for compliance with resp. the DVD-Video and DVD+RW specifications.
28. NCS disc target : DVD-Video contents on a DVD+RW disc. Physical & logical data are checked for compliance with resp. the DVD+RW and DVD-Video specifications.
29. BEE cartridge target : DVD+RW Video logical data recorded in BEE image format. Some physical aspects (e.g. sector header data) are not checked.

30. These options are mainly intended to specify certain relevant parameters in case these are not provided by valid Cross Check data retrieved from other data files on a DVD+RW disc, e.g. VRMI, VMGI or VTSI.
31. With this option the type of audio in a multiplexed stream can be specified. Different types of audio in a multiplexed stream, i.e. different per audio stream, can not be specified with this option. However, this can always be specified in a script file.
32. This option activates logging of the stream's buffer dynamics log to a "gnuplot" compatible ASCII file, which can be used for numerical or graphical buffer analysis. ["gnuplot" is a public domain tool to create graphical representations of data tables; Also MS Excell could be used]
33. VBV-Buffer analysis is of marginal use, since DVD(+RW)-Video do not support the use of the Video ES VBV-Buffer model.
34. Shows a short explanation of every valid command line option.
35. Logs the DVD+RW Video Verifier log file header with a.o. the program's version information.
36. Because parsing is not started at the actual start of the stream, important or even necessary stream data may not be available and unexpected or unjustified error messages are likely to be generated and should be ignored. Variable length coded data (e.g. MPEG or AC-3 audio) may well be out of sync, causing parser syntax errors and in an exceptional case a program crash. As a consequence, this option must be used carefully and is only intended to allow accelerated verification of a particular part of really problematic streams. It is strongly recommended to specify the start sector of a VOBU and preferably a Cell, because then side effects are limited to the minimum. Some checks will be disabled though (cf. 11.8 Disabled Checks in case of Missing Stream Start). Needless to say that the stream's buffer verification is no longer valid.
37. In case of a PS stream, only the pack/packet numbers and stream position of the top layer will reflect the specified start position; PES and SEQ layers will have their packet numbering and stream position starting from 0!
38. Only available for ES stream input.
39. Logical sector numbers (LSN) in case of disc(image) input. Relative sector numbers (RLBN) in case of VOBS file input. Pack numbers in case of VOBS or PS file input. Packet numbers in case of PES input.
40. Not to be used for files without AV contents, such as navigation data files (VRMI, VMGI or VTSI)
41. Illegal settings (e.g. beyond the file size) are ignored.
42. Only relevant in case of multi-channel MPEG-2 Audio ES input consisting of a base and extension stream. This option allows specifying the file name *extfile* of the extension stream, while the specified input file name *filename* contains the base stream data.
43. In fact the specified log file will eventually only contain the generic, non-data file related verification logging, such as program and system errors, Lead-in, Lead-out, disc lay-out and file systems contents dump and error reporting. DVD+RW Video data file related contents dump and error reporting will be written to a specific log file, one for each data file (cf. 5.4 Log Files)
44. The (input) file name is only required in case of non-disc input; in the latter case, it is only used to derive the base name for output files.

45. To better handle Bad Spot Write Errors on a DVD+RW disc and attempt to avoid most of the unjustified errors it may generate, this option allows to specify an alternative parsing control mechanism (cf. 11.1 VTSI Cell Data Controlled Parsing and 11.2 VR Play List Controlled Parsing) which may bypass or jump over the part of the bit stream containing the bad spots. (cf. 12.1 DVD+RW Disc Bad Spots)
46. Via this setting the user can force the use of either the original navigation (VMGI or VTSI) or recording (VRMI) data recorded in the “IFO”-files, or the backup version of this data recorded in the “BUP”-files for cross checking with other data on a disc (image). Selecting IFO data may be useful when the verifier has detected that the IFO data is unreliable or corrupt and automatically reverts to using the backup data. Through this option the user can overrule this and force the use of the (possibly incorrect) IFO data. Specifying the BUP allows the user to discard the original IFO data e.g. because it is bad but has not been detected as such by the verifier, and make the verification process use the BUP data for cross verification. Note that it applies to all IFO/BUP files at once!
47. In some cases, e.g. when a possible Bad Spot Write Error has been detected on a disc (cf. 12.1 DVD+RW Disc Bad Spots), verification will be aborted to prevent loads of unjustified errors because undefined data is being parsed. This option allows to avoid premature verification termination e.g. in case the detection is not correct (since bad spots can not be detected with absolute certainty) or for further analysis of the bit stream data,

3.8 SCRIPT FILE INTERFACE

3.8.1 Script File

With the –**D***scriptfile* option a script file can be passed to the verifier in which, besides most command-line options, also more advanced options can be specified. If an option is specified on both the command line and in the script file, then the latest (scanning from left to right, top to bottom) specified options take precedence.

The syntax for the script file is given below.

Note that,
- *t*+ means that *t* must occur 1 or more times.
- *u* | *v* means that either (not both) the keyword *u* or *v* can be used.
- [*x*] means that *x* is optional.
- < *y* > means that *y* is a logical representation of a class of keywords.
- (*z*) means that *z* is a specification of one of the non-terminals in the script syntax.
- The order of the options is irrelevant, except for **inputfile** and **start_verification**.
- If the same option is specified more than once (by mistake) the last instance of that option will be used.
- *Italic text* is comment
- Gray text: this feature is not yet supported.

A script file must always begin with an input specifier. Any line starting with '!' will be seen as comment. The following lines are valid specifications for the inputfile specifier:

showdir disc <driveletter>
inputfile disc <driveletter> **udf** | **iso** [<subset>]

showdir discimg <filename>
inputfile discimg <filename> **udf** | **iso** [<subset>]

inputfile ps <filename> [**perform_xcheck**]
inputfile pes <filename>
inputfile mpeg2_video <filename>
inputfile mpeg2_audio <filename>
inputfile private1 <filename>
inputfile private2 <filename>
inputfile mpeg1_video <filename>
inputfile mpeg1_audio <filename>
inputfile vrmi <filename> [**generate_xdata**]
inputfile vmgi <filename> [**generate_xdata**] [**perform_xcheck**]
inputfile vtsi <filename> [**generate_xdata**] [**perform_xcheck**]
inputfile pgci <filename>
inputfile spu <filename>
inputfile ac-3 <filename>
inputfile lpcm <filename>
inputfile pci <filename>
inputfile dsi <filename>
inputfile vobs <filename>+ [**perform_xcheck**]

<filename> = name of the input file (preceded by full path if not in current directory)

<subset> = **files** [**generate_xdata**] [**perform_xcheck**] **all**
| **files** [**generate_xdata**] [**perform_xcheck**] <filename>+
| **vmg** [**generate_xdata**]
| **vts** <vtsnr> [**generate_xdata**] [**perform_xcheck**]
| **vobs** <vtsnr> [**generate_xdata**] [**perform_xcheck**]
| **lead-in**
| **lead-out**
<vtsnr> = 1 .. 99

A disc (image) is a collection of files, which is described by a file system. A file is a collection of sectors. Sectors also contain meta data, with information about the sector itself (sector headers).
The verifier can be instructed to verify the following:
- *with <subset> =* **files all**, *a whole disc (image) will be verified, or*
- *with <subset> =* **lead-in**, *the lead-in from a disc (image) will be verified, or*
- *with <subset> =* **lead-out**, *the lead-out from a disc will be verified, or*
- *with <subset> =* **vmg**, *the VMG IFO and menu VOB files from a disc (image) will be verified, or*
- *with <subset> =* **vts** <vtsnr> , *like* **vmg** *+ the VTS IFO and menu VOB files from a disc (image) will be verified, or*
- *with <subset> =* **vobs** <vtsnr> , *like* **vts** *+ the VTS title VOB files from a disc (image) will be verified, or*
- *with <subset> =* **files** <filename>+ ,*the specified files from a disc (image) will be verified.*

perform_xcheck *can be specified to cross-check settings or parameters between different streams (e.g. VMGI and VOBs or between VMGI and VTSI). These cross-checks however need data, stored in a xdata-file. This xdata-file will be generated (or updated when it already exists) with the* **generate_xdata** *option.*

The **vobs** *input type can be used to verify a title VOBS which is larger than 1GB (and thus consists of several VOBS files). The VOBS files as specified in the file list will be concatenated in order of appearance.*

The following specifiers may follow the input specifier and are optional for the script file:

[**start_verification** **at_pack** | **at_packet** | **at_byte** <N>]
(where <N> is a natural number,
at_pack and at_packet are valid for PES and PS input,
at_byte is only valid for ES input)

[**outputdir** <path>]
(where <path> is the full pathname (ending with a slash) of the directory for the logfile and crosscheck file)

The following options may appear in any order:

[**log_progress**]

[**logfile** <filename>]
This option writes all output to file <filename>.

[**xcheckfile** <filename>]
This option specifies file <filename> to be used to store the cross check data.

[**max_error** <N>]
When max_error is specified, all errors are reported at most <N> *times. (default is 1 error)*
Remark: Checks that are similar for e.g. audio and video data will be referenced by the same error number. If more streams are multiplexed into one VOB, these errors may be suppressed for one stream if the maximum error count was already reached for another stream of the VOB.

[**ignore** <mesg>+]
(where <mesg> = **informations** | **oddities** | **recommendations** | **errors** | **warnings | syntax_errors** | **system_errors** | **pipe_errors** | <N>)
This option can be used not to display a certain type of error or a specific error number.

[**gnuplot** **std_buf**]
Outputs the P-STD buffer contents to a gnuplot-compatible ascii-file, which afterwards can be viewed with gnuplot. (The program 'gnuplot' is a public domain tool).

[**linewidth** <N>]
With this option the maximum linewidth of the output can be set.

[**skip_parsing** <skiplevel>+]
(where <skiplevel> = **PS_data** | **PES_data** | **VES_data** | **mblock_data** | **AES_data** | **AMC_data** | **padding_data** | **private_data** | **other_stream_data** | **lead-in** | **sector** | **vrmi** | **vmgi** | **vtsi** | **pgci** | **priv1** | **spu** | **pxd** | **lpcm** | **ac-3** | **priv2** | **pci** | **dsi** | **nav_command**)

VES_data = Video Elementary Stream data
AES_data = Audio Elementary Stream data
AMC_data = Audio MultiChannel data

[**skip_verification** <level>+]
(where <level> = **iso_filesystem** | **udf_filesystem** | **filesystem** | **pack** | **system_header** | **pes_packet** | **pes_data** | **sequence_header** | **gop_header** | **picture** | **slice** | **macroblock** | **block** | **frame** | **frame_header** | **sample_data** | **lead-in** | **lead-out** | **sector** | **vrmi** | **vmgi** | **vtsi** | **pgci** | **priv1** | **spu** | **pxd** | **lpcm** | **ac-3** | **priv2** | **pci** | **dsi** | **nav_command**)

filesystem = iso_filesystem + udf_filesystem

[**dump** [<Ns> [<Ne>]] <level>+]
Dump can be used to output the information of one or more specified levels in readable ASCII-format. <Ns> *is the first Pack which will be dumped,* <Ne> *is the last. When* <Ns> *and/or* <Ne> *are omitted all AU's will be dumped.*
(where <level> = **iso_filesystem** | **udf_filesystem** | **filesystem** | **pack** | **system_header** | **pes_packet** | **pes_data** | **sequence_header** | **gop_header** | **picture** | **slice** | **macroblock** | **block** | **frame** | **frame_header** | **sample_data** | **lead-in** | **lead-out** | **sector** | **vrmi** | **vmgi** | **vtsi** | **pgci** | **priv1** | **spu** | **pxd** | **lpcm** | **ac-3** | **priv2** | **pci** | **dsi** | **nav_command** | **nav_hexdump** | **nav_disasm**)

[**context_dump** <level>+]
If an error is reported, this option dumps the information of the specified levels immediately preceding the error position.

[**appl_checks** **disabled** | **enabled**]
Enables/disables the application specific DVD constraints verification on MPEG-data. Default the verification is enabled.

[**video_coding_mode** **mpeg1** | **mpeg2**]
When the input-stream contains no navigation-data (e.g. VTSI or VMGI) or has no way of determining the video-coding-mode (e.g. cross-data file) this option can be used to specify mpeg1 or mpeg2.

[**audio_coding_mode** <stream_nr> **mpeg1** | **mpeg2**]+
(where <stream_nr> is the number of the stream, 0..7)
As video_coding_mode, but audio-coding-mode can be specified for each audio-stream.

[**tv_system** **pal** | **ntsc**]

[**display_mode** <N>]
N = 0..3:
0: Both Pan-scan and Letterbox
1: Only Pan-scan
2: Only Letterbox
3: reserved for Aspect-ratio 4:3

[**aspect_ratio** <N>]
N = 0, 3:
0: 4:3
3: 16:9

[**source_picture_resolution** <N>]
N = 0..3:
0: 720 x 480 (ntsc), 720 x 576 (pal)
1: 704 x 480 (ntsc), 704 x 576 (pal)
2: 352 x 480 (ntsc), 352 x 576 (pal)
3: 352 x 240 (ntsc), 352 x 288 (pal)

[**audio_channels** <stream_nr> <N>]
stream_nr = 0..7
N = 1..8

[**audio_app_mode** <stream_nr> **karaoke** | **surround**]
stream_nr = 0..7

[**dynamic_range_control** <stream_nr> **on** | **off**]
stream_nr range: 0..7

[**ignore_scrambling**]
When the parser encounters a PES_packet with the PES_scrambling_control set to something other than '00', the parser skips all PES packet payload, because it is scrambled. This payload cannot be unscrambled by the DVD verifier. When the PES_scrambling_control is inappropriately set to something other than '00', this script file option can be used to override the PES_scrambling_control flag, enabling the DVD verifier to parse the PES packet payload normally.

3.8.2 Graphical representation of the Script file syntax

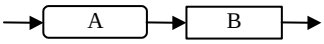
Explanation of symbols:

A = Terminal symbol: type this word exactly as specified.

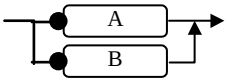
A = Non-terminal symbol: this symbol consists of other terminal and/or non-terminal

For a correct syntax of the script, just follow the arrows. The following constructions are used:

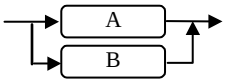
AB (A followed by B)



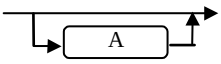
A|B (A xOR B)



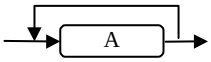
A|AB|B (A OR



[A] (A Optional)

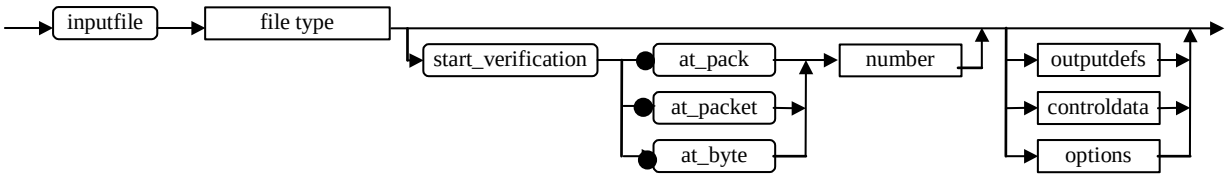


A+ (A 1 or more times)

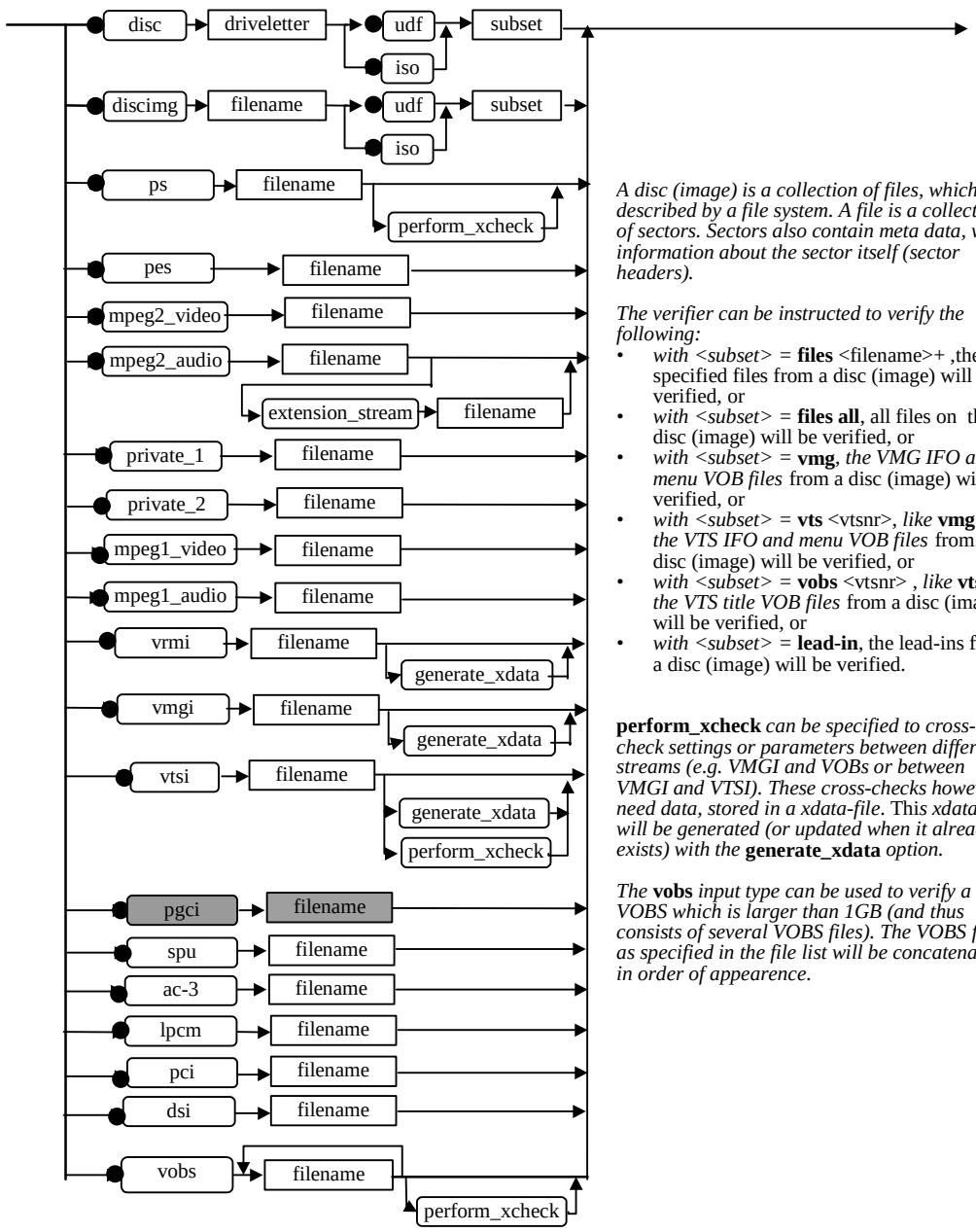


The syntax for the script allows multiple specifications of the same option with different arguments. If an option is specified twice the last specification will be operative.

Scriptfile:



File type:



A disc (image) is a collection of files, which are described by a file system. A file is a collection of sectors. Sectors also contain meta data, with information about the sector itself (sector headers).

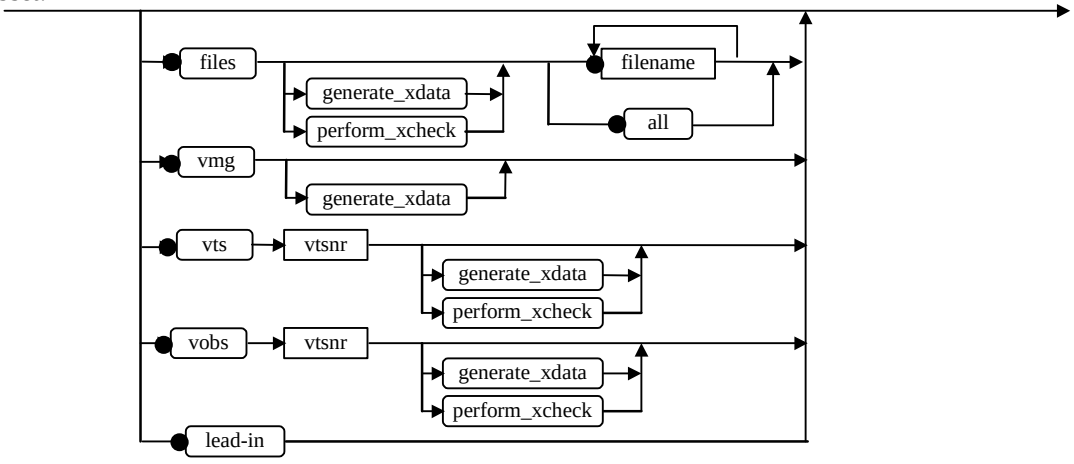
The verifier can be instructed to verify the following:

- *with* <*subset*> = **files** <filename>+ ,the specified files from a disc (image) will be verified, or
- *with* <*subset*> = **files all**, all files on the disc (image) will be verified, or
- *with* <*subset*> = **vmg**, *the VMG IFO and menu VOB files* from a disc (image) will be verified, or
- *with* <*subset*> = **vts** <vtsnr>, *like* **vmg** + *the VTS IFO and menu VOB files* from a disc (image) will be verified, or
- *with* <*subset*> = **vobs** <vtsnr> , *like* **vts** + *the VTS title VOB files* from a disc (image) will be verified, or
- *with* <*subset*> = **lead-in**, the lead-ins from a disc (image) will be verified.

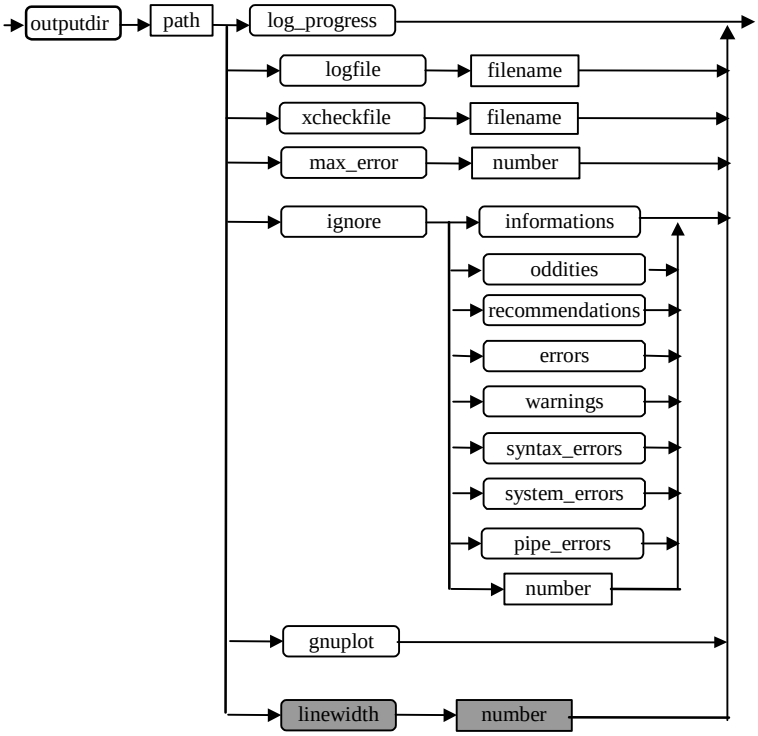
perform_xcheck *can be specified to cross-check settings or parameters between different streams (e.g. VMGI and VOBs or between VMGI and VTSI). These cross-checks however need data, stored in a xdata-file*. This *xdata-file will be generated (or updated when it already exists) with the* **generate_xdata** *option.*

The **vobs** *input type can be used to verify a title VOBS which is larger than 1GB (and thus consists of several VOBS files). The VOBS files as specified in the file list will be concatenated in order of appearence.*

Subset:



outputdefs:



***Outputdir** : The full pathname (ending with a slash) of the directory for the logfile and crosscheck file*

***Xcheckfile**: This option specifies the cross check data <filename>*

***Logfile**: This option writes all output to file <filename>*

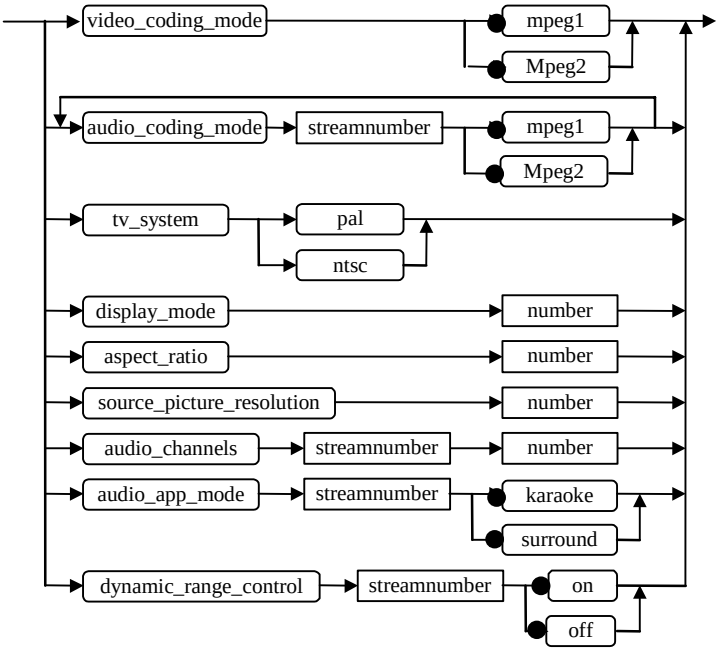
***Max_error:** When max_error is specified, all errors are reported at most <N> times. (default is 20 errors)*

***Ignore**: This option can be used not to display a certain type of error or a specific error number.*

***Gnuplot:** Gnuplot outputs the P-STD buffer contents to gnuplot-compatible ascii-file, which afterwards can be viewed with gnuplot.*
(The program 'gnuplot' is a public domain tool).

***Linewidth:** With this option the maximum linewidth of the output can be set.*

controldata:



***Video_coding_mode:** When the input-stream contains no navigation-data (e.g. VTSI or VMGI) or has no way of determining the video-coding-mode (e.g. cross-data file) this option can be used to specify mpeg1 or mpeg2.*

***Audio_coding_mode:** As video_coding_mode, but audio-coding-mode can be specified for each audio-stream.*

***Display_mode:** N = 0..3:*
0: Both Pan-scan and Letterbox
1: Only Pan-scan
2: Only Letterbox
3: reserved for Aspect-ratio 4:3

***Aspect_ratio:** N = 0, 3:*
0: 4:3
3: 16:9

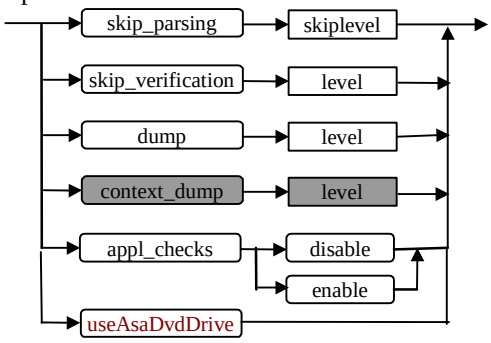
***Source_picture_resolution:** N = 0..3:*
0: 720 x 480 (ntsc), 720 x 576 (pal)
1: 704 x 480 (ntsc), 704 x 576 (pal)
2: 352 x 480 (ntsc), 352 x 576 (pal)
3: 352 x 240 (ntsc), 352 x 288 (pal)

Audio_channels:
stream_nr = 0..7
N = 1..8

Audio_app_mode:
stream_nr = 0..7

Dynamic_range_control:
stream_nr range: 0..7

options:

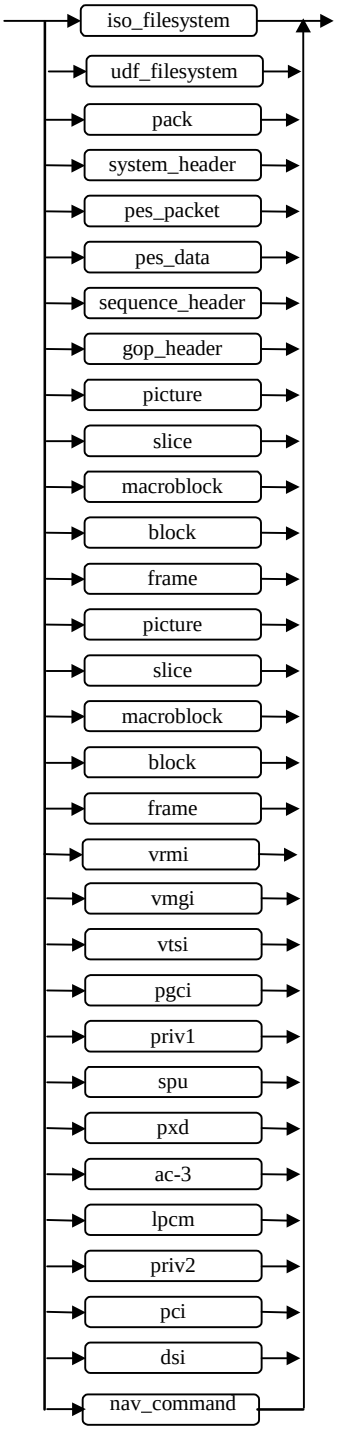


Dump: *Dump can be used to output the information of one or more specified levels in readable ASCII-format.*

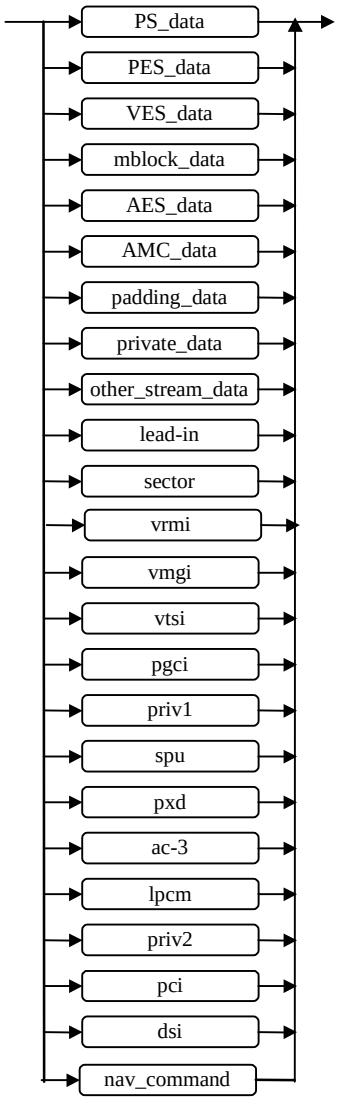
Context_dump: *If an error is reported, this option dumps the information of the specified levels immediately preceding the error position.*

Appl_checks: *Enables/disables the constraints verification on MPEG-data. Default the verification is enabled.*

level:



skiplevel:



3.8.3 Example Script File

```
inputfile discimg 0id_NC04_rev00.toc
udf
files generate_xdata perform_xcheck
all
log_progress
logfile 0_NC04_rev00.log
xcheckfile 0_NC04_rev00.xcheck
skip_parsing mblock_data
max_error 20
tv_system ntsc
```

3.9 SCRIPT FILE VS. COMMAND LINE INTERFACE

All command-line options may be combined with the use of a script file.

As a general rule, in case of conflicts between user input settings either from the command-line or a script file, the last specified settings overrule any preceding setting.

In principle the command line and script file interface offer the same functionality.

The difference is primarily in their use. The script file interface provides a way to preserve the settings for a particular verifier run to reproduce it easily. Apart from this the script file interface is also more user friendly, because the options are specified by keywords rather than the sometimes cryptic or illogic command line characters.

But there are some differences though, which are listed here.

3.9.1 Options only available in the command-line interface

The following options can not be specified by the script file interface:

-**G***type*		**Input stream type**
	F	(UDF + ISO) File System(s) "on file"
	M	BEE cartridge
-**X***subset*		**Specify data subset**
	f	File system(s) data
-**s***level*		**Skip verification** of
	^	UDF 2nd anchor point (AVPD)
-**d***level*		**Generate dump** for the levels:
	b	bit settings
	91	Scan non-zero sectors and hex dump contents
	92	Scan sectors and hex dump their contents
-**C**		**Disable cross-checking**
-**M***type*		**Specify target medium type**
	R	DVD+RW disc
	O	DVD-ROM disc
	N	NCS disc
	E	BEE cartridge
-**P***type*		**Specify parser control method**
	C	(VTSI) Cell controlled
	V	(VRMI) VR Play List controlled
-**U***type*		**Specify IFO or BUP data to use**
	I	Original (IFO)
	B	Backup (BUP)
-**g**		Overrule premature program abort
-**D***scriptfile*		Use script file *scriptfile*

3.9.2 Options only available in the script file interface

The following options can not be specified by a command-line option:

[**perform_xcheck**]
[**generate_xdata**]

[**xcheckfile** <filename>]

[**display_mode** <N>]
[**aspect_ratio** <N>]

[**audio_channels** <stream_nr> <N>]
[**audio_app_mode** <stream_nr> **karaoke** | **surround**]
[**dynamic_range_control** <stream_nr> **on** | **off**]

[**ignore_scrambling**]

4 INPUT FILE(TYPE)S

This section describes the format of input stream files accepted by the DVD+RW Video Verifier.

In principle all input stream files are simple binary files.

However there are some special case, either because of the file contents, or because the input 'file' is actually a file SET. These are all described in more detail below.

4.1 DDP DISC IMAGES

DDP (cf. [DDP]) is the de facto standard format for CD / DVD data exchange between authoring studio's and replicator sites, in which case the data is effectively exchanged on tape medium.

This is actually an input file set, consisting of 3 files per layer of the disc:

- an "ID" file describing the disc image contents and the data it contains
- a "control" file containing 1 Lead-in Control Data Zone ECC-block of 16 sectors, of which one is the important PFI data sector
- a large data file containing the complete Data Zone of a disc (image)

In case of an Opposite track path dual layer disc, no control file for the 2nd layer is present, since there is only one lead-in area on the disc.

However, DVD+RW Video only defines single layer discs.

The actual sector sizes used on the disc image's control and data files is described in the ID file. This may vary from full 2064-byte 'raw' sectors, over 2054-byte sectors to 2048-byte purely payload, 'net' sectors. Dependent on the sector size, the complete or parts of sector header data is not present, and so must not be parsed & verified.

In case a disc image file consists of incomplete sectors (sector size < 2064 bytes), the current verifier's disc image API read routine completes the sector data to the full 2064 bytes, computing the contents of the missing bytes using the (possibly user specified) information it has on the disc image.

Apart from completing sector data, the current DDP disc image API even discards the actual sector header data as it is retrieved form the disc image files and replaces it by its own computed data. This makes it actually useless to verify this data since it is no longer the original disc image data.

Since the control file actually contains only 1 ECC block or 16 sectors, i.c. the first Control Data block (cf. [DVD+RW]), no complete parsing & verification of the Lead-in is possible. Only this single (out of 192 identical copies) Control Data block can and will be verified (unless disabled by the user). But no DIZ data with the important FDCB is present in the disc image data !

4.1.1 DDP format

cf. [DDP]

In this case, the file set complies with "0_"-prefix file naming convention as it was used in the early of DVD days:

	name (layer 0)	**name** (layer 1)
ID file	0id_<base name>.toc	1id_<base name>.toc
control file	0lc_<base name>.toc	1lc_<base name>.toc
data file	0_<base name>	1_<base name>

- <base name> can be any valid alpha numerical text string, not containing spaces. This allows differentiation between multiple disc images.
- The listed files contain the data for layer 0. In case of a dual layer, the 2nd layer's data is stored in a file set using a "1" prefix, instead of "0".
- The "1_"-prefix part of the disc image cannot be verified separately from the "0_"-prefix. It is automatically logically appended to the 1st layer during verification.

4.1.2 DVD CMF format

cf. [CMF]

	name (each layer)
ID file	DDVID.DAT
control file	CONTROL.DAT
data file	IMAGE.DAT

- Note that this format does not allow for differentiation between multiple images, since the file names are fixed.
- In case of a dual layer disc (image), the 2nd layer file set has exactly the same name ! (As a consequence, these must be stored in different directories on hard disk)

4.2 BEE DISC IMAGES

A BEE is a removable hard disk used to emulate the behaviour of the actual DVD+RW recorder basic engine. It was used during the development of the DVD recorder.

The BEE drive contains all logical data of a DVD+RW disc as it could have been created by an actual DVD+RW player, but actually generated by the emulator device. The data of such a BEE "image" is formatted as a set of predefined files:

filename	purpose
I0_dvdvr.dat	1st part of Lead-in (19840 sectors / 39680 kB) with data before Control Data
I1_dvdvr.dat	2nd part of Lead-in (3584 sectors / 7168 kB) with Control Data
O0_dvdvr.dat	Lead-out data (19840 sectors / 39680 kB)
D0_dvdvr.dat	1st part of Data Zone (2 Gbyte)
D1_dvdvr.dat	2nd part of Data Zone (2 Gbyte)
D2_dvdvr.dat	3rd part of Data Zone (remainder)

In fact, any file directory containing this file set will be handled as a BEE drive image by the verifier.

All files contain only sector payload data and so actually consist of **2048-byte sectors**!

Note that not one of these files can individually serve as a valid input file to the verifier. The set has to be complete!

4.3 FILE SYSTEM(S) 'ON FILE'

To allow 'early' verification of file system data generator software, the verifier has been extended with the option to accept file system data without the context of an actual disc or disc image for verification.

Therefore, the file system data, as if extracted from the disc (image) context, has to be provided on a (simple binary) file by itself.

This file must contain ALL (UDF + ISO) file system data, i.e. the complete part of the Data Zone with the file system (s) data, but organised in **2048-byte sectors**.

5 OUTPUT FORMAT

This section describes the format of messages generated by the DVD+RW Video Format Verifier and the output as a result of the dump options.

5.1 ERROR REPORT

5.1.1 Error Messages

A message is generated if a check results in a violation. Such a message consists of the following parts:

- context of the check (application specifier),
- classification of the message (see),
- message number (see),
- Reference to the table or section of the standard to which the message applies (see),
- message descriptive text,
- position in current and higher stream layers where the violation was detected.

Here are a few examples of error messages:

```
>>> [DVD] INFORMATION 4216  (ref. DVD-3 4.2.1 / BP 212) :
   VTSI_MAT: Time Map Table was found in the VTSI
for VTSI at byte 212 bit 0

>>> [DVD] ERROR 5603  (ref. DVD-3 4.1.5-3) :
   Cross Check for 'VTS_SPST_Ns->Number_of_Sub_picture_streams' failed
         for comparison between VTSI_MAT(value 0x0001) and VMGI->VTS_ATRT(value 0x0000)
for VTSI at byte 596 bit 0

>>> [DVD] ERROR 5614  (ref. DVD-3 4.2.2) :
   PTT_Ns from VMGI (4) not equal to number of PTT_SRP (3)
         in TT 7 from VTS 1
for VTSI at byte 2216 bit 0

>>> [DVD] INFORMATION 4408  (ref. DVD-3 4.3.2 (1)) :
   PGC_GI.PGC_CNT: PGC (#1) without VOB detected
for PGCI at byte 12322 bit 1
VTSI at byte 12322 bit 1
```

5.1.2 File Error Summary

Furthermore, at the end of the verification run, an error summary is added, with the total number of errors for each error class (see) and the total number of each specific error listed per number.

Here is an example:

```
**************************************************************************
*
*       ERROR SUMMARY :
*
*       2 Informations
*       No Recommendation violations
*       No Oddities
*       No Warnings
*       No Syntax errors
*       2 Errors
*       No System errors
*
**************************************************************************
**************************************************************************
*
*       DETAILED SUMMARY :
*
*       Information 4408 :   1
*             Error 5614 :   1
*             Error 5603 :   1
*       Information 4216 :   1
*
**************************************************************************
```

5.1.3 Disc Error Summary

This is currently only generated by the command-line version of the verifier.

At the end of a complete disc or disc image verification, a summary of all errors found on the complete disc(image), i.e. on all files it contains, is generated.

It is stored in the 'base' log file (cf. 5.4 Log Files).

Here is an example:

```
*****************************************************************************
*
*       DISC MAJOR MESSAGES SUMMARY :
*
*   Lead-in:      No messages reported
*   Lead-out:     No messages reported
*   File Systems:              1 Warning
*                             27 Errors
*   /VIDEO_TS/VIDEO_TS.IFO:    1 Recommendation violation
*                              3 Errors
*                              1 System error
*   /VIDEO_TS/VIDEO_TS.BUP:    1 Recommendation violation
*                              3 Errors
*                              1 System error
*   /VIDEO_TS/VTS_01_0.IFO:   15 Errors
*                              1 System error
*   /VIDEO_TS/VTS_01_0.BUP:   15 Errors
*                              1 System error
*   /VIDEO_TS/VTS_01_1.VOB:  233 Warnings
*                          12851 Errors
*                              1 System error
*
*****************************************************************************
```

```
**************************************************************************
*
*         DISC DETAILED MESSAGES SUMMARY :
*
*    Lead-in:
*         Information 6305 :    1
*                Error 5032 :    2
*                Error 5031 :    6
*                Error 5030 :    1
*
*    File Systems:
*                Error 6919 :    3
*                Error 6924 :    1
*                Error 6905 :    2
*                Error 6901 :    1
*                Error 6955 :   10
*                Error 6956 :   10
*            Warning 5961 :    1
*
*    /VIDEO_TS/VIDEO_TS.IFO:
*                Error 6532 :    1
*                Error 6525 :    1
* Recommendation violation 6520 :    1
*                Error 6517 :    1
*       Information 4071 :    1
*      System error 5601 :    1
*
*    /VIDEO_TS/VIDEO_TS.BUP:
*                Error 6532 :    1
*                Error 6525 :    1
* Recommendation violation 6520 :    1
*                Error 6517 :    1
*       Information 4071 :    1
*      System error 5601 :    1
*
*    /VIDEO_TS/VTS_01_0.IFO:
*              Error 6594 :    1
*              Error 6593 :    1
*              Error 6591 :    1
*              Error 6590 :    1
*              Error 6587 :    1
*              Error 6586 :    1
*              Error 6630 :    1
*              Error 6613 :    1
*              Error 6610 :    1
*              Error 6602 :    1
*              Error 6564 :    1
*              Error 6563 :    1
*              Error 6557 :    2
*              Error 6552 :    1
*      Information 4216 :    1
*    System error 5601 :    1
*
*    /VIDEO_TS/VTS_01_0.BUP:
```

```
*            Error 6594 :    1
*            Error 6593 :    1
*            Error 6591 :    1
*            Error 6590 :    1
*            Error 6587 :    1
*            Error 6586 :    1
*            Error 6630 :    1
*            Error 6613 :    1
*            Error 6610 :    1
*            Error 6602 :    1
*            Error 6564 :    1
*            Error 6563 :    1
*            Error 6557 :    2
*            Error 6552 :    1
*     Information 4216 :    1
*    System error 5601 :    1
*
*    /VIDEO_TS/VTS_01_1.VOB:
*            Error 1130 : 1198
*
*         Warning 6075 :    1
*           Error 5834 :    2
*          Oddity 4529 :    2
*         Warning 6071 :  167
*           Error 6782 :   14
*           Error 4535 :  304
*         Warning 4536 :   65
*           Error 5801 :  182
*           Error 6787 : 1788
*           Error 6786 : 3456
*           Error 6781 : 4999
*           Error 6769 :  453
*           Error 6751 :  453
*           Error 4617 :    1
*     Information 6701 :  453
*           Error 5803 :    1
*     Information 3012 :    1
*     Information 3011 :    1
*     Information 6025 :    1
*    System error 5601 :    1
*
```

The first list shows the total error count per error class. However only the actual non-compliances and possible errors are taken into account here; Information messages, oddities and non-data related errors are ignored.
The 2nd list gives per file the number of occurrences of each error. The errors are listed in order of their first occurrence.

5.2 CONTENTS DUMP

When the dump option is enabled, token descriptors are logged to the output.

For each level for which a dump is enabled all tokens and their values will be printed. Here is an example:

VRMI is at VRMI stream position 0.

```
VRMI_GI :

(    0:   0:0) VRM_ID                    'DVDVRMANAGER'
(    0:  12:0) … 16 reserved bytes skipped
(    0:  28:0) VRMI_EA                   0x00000010  (16)(RLBN)
(    0:  32:0) VERN                      16
(    0:  34:0) … 30 reserved bytes skipped
(    0:  64:0) DSC_ST.TV_system                0
(    0:  64:2) DSC_ST.reserved           0
(    0:  65:0) … 6 reserved bytes skipped
              LAST_DATE (yyyy/mm/dd):    2000/09/02 (Saturday)
(    0:  71:0) Year (yyyy)               2000
(    0:  73:0) Week                      6 (Saturday)
(    0:  73:3) Month(mm)                 09
(    0:  74:0) reserved                  0
```

At the start of each important access unit encountered in the stream, a small header is printed. This header contains all relevant position information for this AU, such as:

- type of the AU
- its sequential number
- position in current and higher layers of the stream

For each token the following information is printed:

(N:B:b) *field_name value* (*info*)

where,

- N gives the number of the AU,
- B gives the byte position relative to this AU,
- b gives the bit position relative to the byte B, where 0 is msb!
- field_name gives the token description,
- value gives the value of the token,
- (info) gives some additional information about the token.

The indentation of the token description increases with each sub level.

5.3 OUTPUT DIRECTORY

All verifier output files are created in the 'current' output directory.

By default this is the same directory as where the input files are retrieved from.

However the user may explicitly specify another output directory, by typing it in or browsing for it in the GUI 'Misc settings' 'Output directory' box, or through the "-**Z**<directory>" command-line option, or using the "outputdir <path>" script file option.

This is even necessary if the input directory is non-writable because it is write-protected or simply because it is actually a non-writable medium, e.g. CD or DVD disc.

5.4 LOG FILES

Via the command line option "-**L**<filename>" the verifier output can be redirected to a log file.

Note that the GUI version of the verifier always generates a set of log files. The user must not enable this nor can he prevent this.

All output is created in the current output directory.

When verifying a single stream file, a matching log file is generated. But when verifying a complete disc (image), not a single but a complete set of log files is generated, one per file being verified.

The naming convention of the generated log files is as follows:

file name	**contents**	**remark**
command-line version: <filename> GUI version: disc.log	Base log file	1,2
leadin.log	Lead-in error report	1
FileSystems.log	File Systems error report	
leadout.log	Lead-out error report	1
vrmi.log and vrmi_bup.log	VRMI verification report and VRMI backup verification report	
vmgi.log and vmgi_bup.log	VMGI verification report and VMGI backup verification report	
vmgm.log	VMG menu VOBS verification report	
vtsi_0x.log vtsi_bup_0x.log where x = [1..3]	VTSI verification report and VTSI backup verification report (for as many titles as present)	4
vtsv_0x.log where x = [1..3]	VTS title VOBS verification report	3, 4

Remarks:

1. These files are only generated in case of disc or disc image input
2. The base log file catches possible initialisation errors, and, if generated, will receive at the end of a disc(image) verification the disc error summary
3. So there is a single log file for possible multiple VTS title VOBS files, conform the file naming "transcription" done by the GUI upon file selection.
4. At most 3 of these log files can be created, since there are max. 3 VTSs on a DVD+RW Video disc.

5.5 CROSS CHECK DATA FILE

To allow cross checking, i.e. verification whether the data stored in different files on a disc(image) is consistent, the data to be verified has to be stored after the 1st file has been parsed & verified, at least until the last file to be cross checked has been handled. This is achieved by storing this information on a single “cross check data” file that is accessed throughout the verification process to store or retrieve data to be cross checked.

This file is a text file (allowing relatively easy visual inspection of the data for additional analysis).

Depending on the version of the verifier (GUI or command-line) this file has a fixed name or not, and is preserved after verification or not.

verifier version	**GUI**	**Command-line / Script file**
file name	temporary name	“dvd_verif_xdata.xcheck”
	chosen by the system	optionally user selectable through the script file option “xcheckfile”
file scope	deleted at the end of verification	overwritten; not deleted
location	temp directory	current output directory

Cross check data is added to this file during the verification of the basic navigation data (IFO) files such as the VRMI, VMGI and VTSI files, in the same sequential order as they are verified. As a consequence, when cross checks are desired, one always has to verify the files in the correct hierarchic order, which is:

1. VRMI: not needing data from any other file
2. VMGI: only using some VRMI data for cross checking
3. VTSI: using both VRMI and VMGI data for cross checking
4. VOBS: cross checks with data from all navigation files

If the order is not maintained, proper cross verification is not possible and, dependent on the cross check data that is missing, the verifier will use some (possibly incorrect) default parameters or will not perform some checks. However, if this happens, it will always be reported by the verifier.

As a consequence, only the command-line version of the verifier allows using previously generated and preserved cross check data, if it is available at the expected location. This allows performing cross checks on a single file without first having to re-verify the files it uses the cross check data of.

6 ADDITIONAL FUNCTIONALITY

6.1 DUMP BIT SETTINGS

A disc's "physical" data, encoded in the sector headers and Lead-in, contains some information about the actual (physical) format of the disc, whether it is a DVD-ROM, DVD-R, DVD-RW, DVD+RW, DVD+R, SACD, etc.

The verifier's functionality has been extended with an option that allows visualising this information. If you use an ASA-lab verification drive (JBE or ASD1) it is possible to dump the bit settings. Go to the GUI Settings dialog, 'Dump options'. Select the 'Bit settings' button or specifying the command-line option "**-db**", the verifier will generate information messages logging the contents (in hex) of the physical data bytes at some critical locations:

>>> [DVD+VR] INFORMATION 6493 (ref. +RW N/A) :
Encoded bit settings at sector PSN $02EEC0 (192192) : $ 04 / xx / xx.

>>> [DVD+VR] INFORMATION 6491 (ref. +RW N/A) :
Encoded bit settings at sector PSN $02F200 (193024) : $ 04 / 01 / 01.

Currently the value of the following bytes is dumped:

Lead-in PFI	byte 0 byte 2
sector header ID (MSB)	byte 0

It is currently dumped at the following locations:

PSN	**Area**
0x2EEC0	start of the Lead-in inner Disc Identification Zone
0x2F200	start of the Lead-in Control Data Zone
(various)	start of each file in the Data Zone

Note that the specified maximum number of messages per error also limits the number this information message will be shown.

For interpretation of these byte values, [DVD+RW] should be consulted.

Remark:
Since this information is part of a disc's "physical data", it is only available and relevant in case this data is actually retrieved by using the special ASALE-made verification drive (cf. Appendix B).

6.2 TARGET MEDIUM

Although the DVD+RW Video Format Verifier has been designed to check the conformance of DVD+RW Video discs data, variants of this format have been defined and used throughout the development of the standard document, test discs and recorder devices.

At least to be able to properly verify test discs with either DVD+RW contents on a replicated DVD-ROM disc, or DVD-Video contents on a DVD+RW disc, the verifier has been extended with the possibility to specify the disc type currently under test. This can be achieved by selecting the correct type in the GUI ‘Misc settings’ ‘Target medium’ or by specifying it through the “-**M**<RONE>” command-line option.

type	**contents**	**carrier**	**remark**
DVD+RW	DVD+RW	DVD+RW	default
DVD-ROM	DVD+RW	DVD-ROM	physical DVD-ROM checks
NCS	DVD-ROM	DVD+RW	logical DVD-Video checks
BEE	DVD+RW	BEE	no sector header checks

Selecting the correct disc type enables and disables the appropriate verifier modules to avoid unjustified error messages.

6.3 SECTOR SCAN

The verifier tool can hex dump the payload of parsed sectors as another means of (low level) data analysis.

Specifying the command-line option "**-d91**" logs for every parsed sector if it is not all-zero:

```
>>> pack          0 (0x000000) non-zero
>>> pack          1 (0x000001) non-zero
>>> pack          2 (0x000002) non-zero
>>> pack          3 (0x000003) non-zero
…
```

Specifying the command-line option "**-d92**" logs for every parsed sector the 2048 bytes payload in hex + character format, e.g.:

```
>>> pack          0 (0x000000)
   0 : 0000 01BA 4400 0400 0401 0189 C3F8 0000  |  <    ╟D       ë °
  16 : 01BB 0012 80C4 E100 E17F B9E0 E8B8 C020  |  < ╕  Ç      ╣αΦ └
  32 : BDE0 3ABF E002 0000 01BF 03D4 0000 0000  |  <╜α:┐α     ┐ ╘
  48 : 0000 0000 0000 0000 0000 0022 7800 00AF  |  <              "x  »
…
```

Combining both options by specifying "**-d912**" logs the sector payload only for non-zero sectors.

Notes:

!!! This option is only available in the command-line version of the DVD+RW Video Verifier.

- This option is only intended to be used for VOBS data, and does not work (properly) for navigation data files.
- This dump is limited by the specified AU dump range.

7 PHYSICAL DATA PARSING & VERIFICATION

Physical data, i.e. data recorded in a DVD+RW disc's Lead-in or sector headers, is only partial available in a DDP / CMF disc image input file set (cf. 4.1 DDP Disc Images) or on an actual DVD+RW disc when accessing it with a drive capable of retrieving this physical data, e.g. an ASALE verification drive (cf. Appendix B Verification Drive).

The following table lists which data elements are present on a disc image, and which only in the final disc (of course everything must be correctly set in an actual disc).

Lead-in data			Valid on a disc image	Verified for a disc image	Remark
DIZ data			**-**	-	5
PFI data			√	√	
sector 0	byte 0	Disc Category & Version Nr.	√	√	
	byte 1	Disc size & max. transfer rate	√	√	
	byte 2	Disc structure	√	√	
	byte 3	Recording density	√	√	
	byte 5..7	Data Zone start address	√	√	
	byte 9..11	Data Zone end address	√	√	
	byte 13..15	reserved	√	√	
Sector header data					
ID	byte 0..2	Sector Number	√	-	2,4
	byte 3	Sector Info	√	-	2,4
IED	byte 0..1		√	-	2,4
CPSI	byte 0..5	reserved	**-**	-	3
EDC	byte 0..3		√	-	2,4

Remarks:
1) May be changed in the future.
2) Only present in case of a "Complete, 2064-byte Record size" sector formatted main data file ("0_<name>, "IMAGE.DAT" or "MAIN.DAT", cf. [DDP] or [CMF] SSM data field).
3) Only present in case of a "2054- or 2064-byte Record size" sector formatted main data file ("0_<name>, "IMAGE.DAT" or "MAIN.DAT", cf. [DDP] or [CMF] SSM data field).
4) In new disc images these data fields may no longer be present (all set to zero), since their value may conflict with the actual state (e.g. sector address or certain bit flags set and not taken into account by the EDC). Therefore sector header data is not checked for disc images.
5) This data can not be stored in the current DDP / CMF format.

Physical Data not present in a disc image is only parsed and verified when verifying an actual DVD+RW disc (and using a special verification drive). When reading from a disc image, these data elements are simply skipped and never verified.

Currently only "2054-byte Record size" control data files (containing the Lead-in data) and "Complete, 2064-byte Record size" image files (containing the actual Data Zone data) are supported (cf. [DDP] or [CMF] SSM data field).

8 ERROR NUMBERS

8.1 ERROR CLASSES

The following error classes are used by the messages:

Error Class	Explanation
INFORMATION	A notable event in the stream (purely informative)
RECOMMENDATION VIOLATION	A violation against a recommendation
ODDITY	An odd situation or inconsistency in the stream
WARNING	A potential cause of errors
SYNTAX ERROR	A violation against the syntax, detected during parsing.
ERROR	A violation against a mandatory requirement
SYSTEM	A verifier error (non DVD+RW)

All SYNTAX ERRORs also print a look-ahead buffer with the next few bytes (max. 4) in the bit pipe. It is represented as follows.

```
>>> message <<<
   [Look Ahead : $00 $01 $B3 $2D  (len : 32 bit) ]
```

The numbers after Look Ahead represent the next bytes from the stream in hexadecimal notation, followed by the number of bits. A maximum of 32 bits (4 bytes) can be printed.

8.2 SPECIFICATION REFERENCES

The references to the MPEG, DVD-Video and DVD+RW specifications used in the messages refer to:
(see 1.3 References)

Message Text	Reference
MPEG-2 Systems	[MPEG-2 Systems]
MPEG-2 Video	[MPEG-2 Video]
MPEG-2 Audio	[MPEG-2 Audio]
MPEG-1 Video	[MPEG-1 Video]
MPEG-1 Audio	[MPEG-1 Audio]
MPEG Video	[MPEG-2 Video] and [MPEG-1 Video]
UDF or ECMA	[UDF] resp. [ECMA]
ISO-9660	[ISO]
AC-3	[AC-3]
DVD	[DVD-Video]
DVD-ROM	[DVD-PHYS]
DVD+RW Video	[DVD+VR]
DVD+RW	[DVD+RW]

When a verifier message contains a double MPEG reference, the first one is always an MPEG1 and the second an MPEG-2 reference. For example, in “(ref. MPEG Systems 2.4.4.2 | 2.5.3.6)”, the first one is a reference to MPEG Systems; section 2.4.4.2 and the second to MPEG-2 Systems; section 2.5.3.6.

8.3 CHECK GROUPS

Check Group		Error number range
System errors and Exit codes		0 - 999
MPEG	MPEG-1 PRS checks	1100 - 1199
	MPEG-1 System header checks	1200 - 1399
	MPEG-1 PES checks	1400 - 1499
	MPEG-1 Sequence header checks	1500 - 1619
	MPEG-1 GOP checks	1620 - 1649
	MPEG-1 Picture checks	1650 - 1749
	MPEG-1 Slice checks	1750 - 1769
	MPEG-1 Macroblock checks	1770 - 1799
	MPEG-1 Video block checks	1800 - 1849
	MPEG-1 Audio checks	1850 - 1949
	MPEG-2 PRS checks	2300 - 2399
	MPEG-2 PES checks	2400 - 2499
	MPEG-2 Sequence header checks	2500 - 2619
	MPEG-2 GOP checks	2620 - 2649
	MPEG-2 Picture checks	2650 - 2749
	MPEG-2 Slice checks	2750 - 2769
	MPEG-2 Macroblock checks	2770 - 2799
	MPEG-2 Video block checks	2800 - 2849
	MPEG-2 Audio checks	2850 - 2899
	MPEG-2 Descriptor checks	2900 - 2999
DVD-Video	System checks	3000 - 3009
	VOB checks	3010 - 3049
	Pack checks	3100 - 3149
	System header checks	3150 - 3199
	Packet checks	3200 - 3249
	PES checks	3250 - 3299
	Private stream checks	3300 - 3349
	Sequence header checks	3350 - 3399
	GOP checks	3400 - 3449
	Picture checks	3450 - 3479
	audio checks	3500 - 3599
	SPU checks	3600 - 3749
	AC3 checks	3750 - 3849
	LPCM checks	3850 - 3899
	VMGI checks	4000 - 4199
	VTSI checks	4200 - 4399
	PGCI checks	4400 - 4499
	PCI checks	4500 - 4599
	DSI checks	4600 - 4799
	NCMD checks	4800 - 4899
	(unused)	4900 - 4949
DVD-ROM	Sector checks	4950 - 4969
	drive errors	4970 - 4989
	DVD+RW specific Sector checks	4990 - 5000
Bigfile	(big file I/O) errors	5000 - 5009
Lead-in	Generic checks	5010 - 5021
	DVD-Video specific checks	5022 - 5025

	DVD+RW Video specific checks	5030 - 5049
file systems	UDF file system checks	5050 - 5499
	ISO-9660 file system checks	5500 - 5599
Cross checks	DVD	5600 - 5949
	File systems	5950 - 6000
DVD+RW	Generic System checks	6001 - 6049
Video	VOB checks	6050 - 6099
	PRS checks	6101 - 6149
	PES checks	6150 - 6179
	SPU checks	6180 - 6199
	PRV checks	6200 - 6249
	Video checks	6250 - 6269
	Picture checks	6270 - 6279
	Audio checks	6280 - 6299
DVD+RW	Physical Data checks	6300 - 6499
	Lead-in / Lead-out DCB checks	6361 - 6389
	Lead-out checks	6400 - 6425
	Lead-in vs. Lead-out Cross checks	6451 - 6459
DVD+RW	Navigation Data checks	6500 - 6799
Video	VRMI checks	6801 - 6899
	Disc Layout checks	6901 - 6949
	File System checks	6951 - 6959
	Cross checks	6960 - 6999
	Content Protection checks	8000 - 8100

8.4 EXIT CODES

The exit codes are reported to indicate a serious problem.

Exit code	Meaning	
1	Unable to create the log file	
2	Unable to open the input file	
3	Unable to open the output file	
4	Unable to access the input file	
5	User abort	
6	Error reading input file	
19	Nr of errors overflow	D
21	Mixed DOS & UNIX slashes (‘/’ and ‘\’ were mixed)	
22	Missing slash in dir name (dir name should end with slash)	
23	Equal file names for input and output files	
31	Illegal input stream type	
32	Input stream does not start with a startcode	
33	Input stream does contain any startcode	
42	Audio seq. does not contain any sync	
53	Trying to read beyond end of file	
60	Internal error, 0 length block passed	D
61	Complete_pack : Unexpected EOI_CODE	
71	Negative position value indicates parsing problem	D
72	Too large position value indicates parsing problem	D
75	Too large bitpip read length, maximum of 32 bits can be read	D
76	Illegal bitpip read length	D

77	Too short bitpip look ahead read position	D
78	Illegal bitpip look ahead read position	D
79	Bit pipe bit offset non zero	D
81	Malloc or Calloc failed	
82	Realloc failed	
85	(VDI) DDP disc image API error	
86	(VDI) DDP disc image API error	
87	(VDI) DDP disc image API error	
92	Illegal command line options	
91	Too many command line options, maximum is 100	
95	Untested feature	
96	Unsupported feature	
98	NULL pointer access	
99	Unexpected exit	
115	No input files have been specified	
116	Unable to open disc image	
117	File not found in the filesystem	
118	SDKA file open error in lead-in	
119	SDKA file write error in lead-in	
120	Linear PCM input (ES) is not supported	
121	Unknown Private-1 (ES) input is not supported	
122	Unknown Private-2 (ES) input is not supported	
123	Unexpected stream input (MPEG1 PES or MPEG1 PS stream)	
126	Unable to open the STD buffer contents log file	
130	Illegal audio type specified for –a command line option	
131	Illegal parameter for –x option	
132	Illegal parameter for –c option	
151	Illegal navigation command type	
170	AC3 decoding state is illegal	
171	AC3 syncinfo illegal decoding state	
172	AC3 bsi illegal decoding state	
173	AC3 audblk illegal decoding state	
174	AC3 aux illegal decoding state	
175	LPCM illegal decoding state	
176	SPU: Illegal decoding state	
177	SPU DCSQT: Illegal decoding state	
178	SPU DCCMD: Illegal decoding state	
179	SPU DCCMD data: Illegal decoding state	
180	SPU PXD inc pix, unexpected run value	
990	Illegal audio input setting	
999	Illegal video input setting	
6009	Illegal file-subset selected for the input settings	
19999	Invalid disc type, when trying to seek a particular sector	

Note: The last column specifies if the exit code is found in the Debug version of the verifier only.

9 COMPLETE ERROR MESSAGE LIST

9.1 SYSTEM CHECKS

These messages are specific verifier system messages. They mostly indicate a problem with the system on which the verifier is running, or with the software tool's internal administration. In principle, these should never occur during 'normal' verification.

>>> [SYSTEM] SYSTEM ERROR 1 :
OPEN_FILE : Can't create log file 'filename' : 'error string'
The system was unable to create the logfile, reported is the filename specified as the logfile as well as the translated error number reported by the system.

>>> [SYSTEM] SYSTEM ERROR 2 :
OPEN_FILE : Can't open input file 'filename' : 'error string'
The system was unable to open the input file, reported is the filename as well as the translated error number reported by the system. This error is reported when:

- The specified script file cannot be opened.
- The specified input file cannot be opened.
- The specified file is not found in the filesystem when the input stream type is a disc or discimage.

>>> [SYSTEM] SYSTEM ERROR 3 :
OPEN_FILE ('function name') : Can't open output file 'filename' : 'error string'
The system was unable to open an output file, reported is the name of the function that tried to open the file, the filename and the translated error number reported by the system. This error is reported when:

- The demux option could not open the output file for the demultiplexed stream.
- The output file for any of the STD, T-STD, VBV buffer dumps could not be opened.

>>> [SYSTEM] SYSTEM ERROR 6 :
FILE_INPUT : Error reading input file : 'filename'
The system encountered an error while reading a file, reported is the filename specified. This error occurs when the number of bytes read from the file does not equal the requested number of bytes.

>>> [SYSTEM] WARNING 10 :
GEM : duplicate client/event link discarded for event 'hexadecimal event number' !
This warning indicates a programming error, the development team should be consulted.

>>> [SYSTEM] WARNING 11 :
GEM : duplicate client/event link for event 'hexadecimal event number': increased priority to 'priority number' !
This warning indicates a programming error, the development team should be consulted.

>>> [SYSTEM] WARNING 12 :
GEM : Event 'hexadecimal event number' to unlink not in the event call-back list !
This warning indicates a programming error, the development team should be consulted.

>>> [SYSTEM] WARNING 13 :
GEM : Client for event 'hexadecimal event number' to unlink not in the client/method call-back list !
This warning indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 19 :
Report module : 'System | MPEG | MPEG-1 | MPEG-2' verification message nr. 'number' is larger than the allowed number of errors ('maximum number').
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 21 :
Specified 'filename type' name ('filename')
mixes UNIX and PC/DOS slashes !!!
When the complete filename is created from the specified output directory and filename, only one type of slashes may be used, i.e. either PC/DOS ('\') or UNIX ('/') type slashes.

>>> [SYSTEM] SYSTEM ERROR 22 :
Specified output directory ('directory name') is not terminated by a slash !
The output directory should be terminated with a slash.

>>> [SYSTEM] SYSTEM ERROR 23 :
Specified input file ('filename') and
log file ('filename') are identical!
The user should specify different filenames for the input file and the log file, which is an output file. This error prevents an endless running task.

>>> [SYSTEM] INFORMATION 30 :
ANALYSE_STREAM : Input stream type probably is a 'streamtype' (found start code 'hexadecimal start code' at byte 'position').
This information reports that the specified input streamtype is probably not correct because another start code was found then specified at the command-line with option –G*type* at a valid position (i.e. No emulated start code) that suggests a different streamtype.
This information message is given when using the command-line verifier with wrong –G*type* parameter.
This information is usually preceded by the SYSTEM ERROR 31.

>>> [SYSTEM] SYSTEM ERROR 31 :
ANALYSE_STREAM : Not a 'streamtype' input stream !!!
The start code belonging to the specified streamtype is not found, the following table show the start codes used by streamtypes:

Streamtype	**Start codes**
PS	0x000001B9, 0x000001BA, 0x000001BB
PES	0x000001BC 0x000001FF
Video ES	0x000000000, 0x00000101 .... 0x000001AF, 0x000001B2, 0x000001B3, 0x000001B4, 0x000001B5, 0x000001B7, 0x000001B8

>>> [SYSTEM] INFORMATION 32 :
ANALYSE_STREAM : Input stream does not start with a start code.
The input stream should start with a start code. Data will be skipped until a valid start code is found in the stream.

>>> [SYSTEM] INFORMATION 33 :
ANALYSE_STREAM : Input buffer does not contain any start code.
The input buffer contains the start of the input stream, its size is dependent on the verification application, but is generally in the range of 32KB to 256 KB. In this buffer, no start code was found, which usually means that the stream is not an MPEG stream. This information is usually preceded by INFORMATION 32.

>>> [SYSTEM] INFORMATION 41 :
ANALYSE_STREAM : Audio seq. does not start with a sync.
This information is reported when the input streamtype is set to Audio ES and the stream did not start with an Audio syncword (0xFFF). Data will be skipped until a syncword is found in the stream.

>>> [SYSTEM] SYSTEM ERROR 42 :
ANALYSE_STREAM : Audio seq. does not contain any sync !
This information is reported when the input streamtype is set to Audio ES and the input buffer did not contain any Audio syncword (0xFFF). The input buffer contains the start of the input stream, its size is dependent on the verification application, but is generally in the range of 32KB to 256 KB. This error is usually preceded by INFORMATION 41.

>>> [SYSTEM] INFORMATION 51 :
ANALYSE_STREAM : Transport stream does not start with a sync.
This information is only applicable for verification applications that support Transport streams and is reported when the stream did not start with an Transport Stream syncword (0x47). Data will be skipped until a syncword is found in the stream.

>>> [SYSTEM] SYSTEM ERROR 52 :
ANALYSE_STREAM : Transport stream does not contain any sync !
This information is only applicable for verification applications that support Transport streams and is reported when the input buffer did not contain any Transport Stream syncword (0x47). The input buffer contains the start of the input stream, its size is dependent on the verification application, but is generally in the range of 32KB to 256 KB. This error is usually preceded by INFORMATION 51.

>>> [SYSTEM] SYSTEM ERROR 53 :
FILIO_READ_FILE : Cannot read beyond the EOF !
The parser needs to read from the file, but the file pointer arrived at the end of the file, meaning no data can be read from the file. This error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 54 :
FILIO_READ_FILE : Parsing completed, but I/O buffer is not empty !
The parser is finished reading data from the file, but the file pointer did not arrive at the end of the file, meaning more data can be read from the file. This error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 60 :
'function name' passed a zero length block of data !
This error reports an error in one of the parser functions. This warning could indicate a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 61 :
parser_input : Unexpected MPEG_PROGRAM_END_CODE !
The parser was expecting a new pack, but instead it encountered an MPEG program end code.

>>> [SYSTEM] SYSTEM ERROR 63 :
parser_input : Input buffer does not contain 'start code type string' start code ! ('hexadecimal start code')
The parser uses an input buffer of a complete pack or complete packet.

>>> [SYSTEM] INFORMATION 64 :
parser_input : Emulated start_code 'hexadecimal start code' !
The parser encountered some bytes in the stream, that emulate a known start code. This could pose a problem if the parser or player should have to recover.

>>> [SYSTEM] INFORMATION 65 :
parser_input : Input buffer too small to contain 1 complete pack. retrieving complete packets...

>>> [SYSTEM] SYSTEM ERROR 66 :
parser_input : 'Packet type string' length ('length') inconsistent with distance between start codes ('length') !
The previous PES_packet or pack was too long.

>>> [SYSTEM] SYSTEM ERROR 67 :

PSI-'table name' parsing : Invalid state 'number'
This error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 69 :
PSI parsing (psi_decode_prog_nr) : Invalid type 'number'
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 71 :
'function name' results in NEGATIVE position value ('value' - 'value') !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 72 :
'function name' results in a TOO LARGE (> 32 bit) position value ('value' - 'value') !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 75 :
Bit pipe error in 'function name' :
Specified length ('value') larger than available ('number') !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 76 :
Bit pipe error in 'function name' :
Specified length ('value') smaller or larger than allowed ('minimum value'..'maximum value') !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 77 :
Bit pipe error in 'function name' :
Specified pipe read position ('value') shorter than allowed ('number') !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 78 :
Bit pipe error in 'function name' :
Specified pipe read position ('value') should be in ['minimum value'..'maximum value'] !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 79 :
'function name' : Detected a non-zero bit offset ('value')
This might indicate a serious parsing problem.
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 81 :
Requested 'C|M'alloc of 'value' bytes FAILED !
This system error indicates the program could not allocate the number of bytes necessary. This could be due to low memory of a programming error. Close some applications and try again.

>>> [SYSTEM] SYSTEM ERROR 82 :
Requested realloc of address 'hexadecimal address' to 'value' bytes FAILED !
This warning indicates the program could not allocate the number of bytes necessary. This could be due to low memory of a programming error. Close some applications and try again.

>>> [SYSTEM] SYSTEM ERROR 91 :
Illegal command-line option !
The user specified an illegal command line option. Check the user manual for all valid command line options and try again.

>>> [SYSTEM] SYSTEM ERROR 92 :
Too many options for CmdLn to hold !
The user specified command line options with more than 100 characters, which is too large. Reduce

>>> [SYSTEM] SYSTEM ERROR 95 :

!!! 'profile or level type' NOT or not sufficiently TESTED !!!
This system error indicates that the specified profile or level is not tested yet. Only main profile, main level is fully tested.

>>> [SYSTEM] SYSTEM ERROR 96 :
!!! 'profile or level type' NOT SUPPORTED or IMPLEMENTED (YET) !!!
This system error indicates that the specified profile or level is not supported yet. Only main profile, main level is fully supported.

>>> [SYSTEM] SYSTEM ERROR 98 :
NULL pointer reference in function 'function name' !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 99 :
! 'error string' !
This system error indicates a programming error, the development team should be consulted.

>>> [SYSTEM] SYSTEM ERROR 101 :
No pack data (mux_rate, SCR) available
A packet cannot be parsed without pack information, which was the case here.

>>> [SYSTEM] SYSTEM ERROR 201 :
complete_t_packet : Stream does not start with a transport_packet sync.

>>> [SYSTEM] SYSTEM ERROR 202 :
complete_t_packet : No transport_packet sync_bytes where expected !

>>> [SYSTEM] ERROR 901 (ref. MPEG-2 Systems 2.4.2.1) :
SCR tolerance exceeded by 'value'

>>> [SYSTEM] ERROR 902 (ref. MPEG-2 Systems 2.4.2.2) :
PCR tolerance exceeded by 'value'

>>> [SYSTEM] ERROR 903 (ref. MPEG-2 Compliance 9.1.3) :
Audio sample frequency tolerance exceeded by 'value'

>>> [SYSTEM] ERROR 904 (ref. MPEG-2 Compliance 9.1.3) :
Video sample rate tolerance exceeded by 'value'

9.2 MPEG CHECKS

These messages relate to generic MPEG checks.

9.2.1 Common MPEG-1 and MPEG-2 checks

9.2.1.1 MPEG PS checks

>>> [MPEG] SYNTAX ERROR 1103 (ref. MPEG Systems 2.4.3.2 | 2.5.3.4) :
Pack too short
The Pack was less than 12 bytes for MPEG-1 streams and 14 bytes for MPEG-2 streams.

>>> [MPEG] SYNTAX ERROR 1104 :
No packs in program stream preceding MPEG_program_end_code
At least 1 pack must precede the MPEG_program_end_code.

>>> [MPEG] SYNTAX ERROR 1105 :
Stream data following MPEG_program_end_code
No data should follow an MPEG_program_end_code.

>>> [MPEG] SYNTAX ERROR 1106 :
Expecting packet_start_code etc. (Look Ahead : 'hexadecimal buffer contents')
The first 4 bytes of the Pack's payload must be a valid start code. The look ahead shows what the first 4 bytes of the payload are. This error is usually due to a recovery action that did not pickup the parsing process at the right place (possible emulated start code).

>>> [MPEG] ERROR 1108 (ref. MPEG systems 2.4.4.1 | 2.5.3.2) :
Program stream is not terminated by an MPEG_program_end_code.

>>> [MPEG] SYNTAX ERROR 1109 (ref. MPEG-1 Systems 2.4.3.2) :
Pack_header marker 0010 expected

>>> [MPEG] SYNTAX ERROR 1110 (ref. MPEG Systems 2.4.3.2 | 2.5.3.4) :
Pack_header marker_bit 'number' is 0
This error is reported when any of the marker bits in the Pack_header is not equal to '1'.

>>> [MPEG] ERROR 1117 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
Pack_header SCR difference is 'difference', should be in 'min' ... 'max' (c1 in 'min' ... 'max')
This error is only reported when the Program stream uses a fixed bitrate. The error reports that the bitrate fluctuates too much to be considered fixed. This error can indicate an erroneous value for the fixed_flag field, in which case this error will be generated very often. It can also indicate an invalid SCR value, in which case the range for a valid SCR difference is given (first range in the error message).

>>> [MPEG] ERROR 1120 (ref. MPEG Systems 2.4.4.2 | 2.5.3.4) :
Pack_header program_mux_rate is 0
The program_mux_rate field in the Pack_header is forbidden to be coded as '0', meaning a muxrate of 0 bytes/sec.

>>> [MPEG] ERROR 1125 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
Pack_header program_mux_rate is 'bitrate' Mbit/s, system_header rate_bound only 'bitrate'
The program_mux_rate from the Pack_header must be less than the rate_bound specified by the system_header.

>>> [MPEG] ERROR 1130 (ref. MPEG Systems 2.4.4.2 | 2.5.3.4) :
Pack_header SCR difference is 'difference', should be at least 'difference'
The difference of two succeeding Pack's SCRs should be at least:

(CLOCK_FREQUENCY * (pack length - 9 {=last byte of SCR})/byterate of previous pack) +
(CLOCK_FREQUENCY * (9 {=last byte of SCR}/ byterate of current pack))

Note: CLOCK_FREQUENCY is 90 KHz for MPEG-1 PS and 27 MHz for MPEG-2 PS

>>> [MPEG] ERROR 1131 (ref. MPEG Systems 2.4.5.2 | 2.7.1) :
Pack_header SCR difference is 'difference' [90kHz ticks], should be at most 'number' seconds (= 'difference').
The difference of two succeeding Pack's SCRs should be at most 0.7 seconds, or (CLOCK_FREQUENCY * 0.7) clicks apart.

Note: CLOCK_FREQUENCY is 90 KHz for MPEG-1 PS and 27 MHz for MPEG-2 PS

>>> [MPEG] ERROR 1141 (ref. MPEG Systems 2.4.6 | 2.7.9) :
Previous pack contains 'value' packet(s), should be <= 'calculated value' for a CSPS-stream
The Pack of a CSPS constrained MPEG Program stream may contain only the number of packets calculated with following formula:

(SCR difference * rate_bound {from system header}) /
(CLOCK_FREQUENCY / 300 {max pack rate} * 400 {bits/sec} * mux_rate_lim)

Note: CLOCK_FREQUENCY is 90 KHz for MPEG-1 PS and 27 MHz for MPEG-2 PS
mux_rate_lim is 5000000 for MPEG-1 PS, 2000000 for MPEG-2 PS when packet_rate_restriction_flag from pack header equals 0 and 4500000 for MPEG-2 PS when packet_rate_restriction_flag from pack header equals 1.

>>> [MPEG] ERROR 1142 (ref. MPEG Systems 2.4.6 | 2.7.9) :
'number' packet(s) in pack preceding EOI-code, max. 'number' allowed
The Pack in which an MPEG_program_end_code exists, may contain a maximum number of packets calculated with following formula:

when rate_bound {in bits/sec}<= mux_rate_lim:
(pack length * 300 {max pack rate} / byte rate

when rate_bound {in bits/sec}> mux_rate_lim:
(pack length * rate_bound {from system_header} * (300 {max pack rate} * 400 {bits/sec} / mux_rate_lim)) / byte rate

Note: mux_rate_lim is 5000000 for MPEG-1 PS, 2000000 for MPEG-2 PS when packet_rate_restriction_flag from pack header equals 0 and 4500000 for MPEG-2 PS when packet_rate_restriction_flag from pack header equals 1.

9.2.1.2 MPEG System header checks

>>> [MPEG] SYNTAX ERROR 1200 (ref. MPEG Systems 2.4.3.2 | 2.5.3.6) :
System_header too short
This error indicates a problem with the header_length field or with the decoding of the P-STD_buffer fields. It is caused by the parser that wants to parse more data than available.

>>> [MPEG] ERROR 1201 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header preceded by 'number' packets in this pack
The System_header may be present in any Pack of a Program stream, but must be located immediately following the Pack header. So no packets may precede the system_header.

>>> [MPEG] ERROR 1202 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
First pack in this stream does not contain a system header

It is mandatory that the first Pack in a Program Stream carries the system header.

>>> [MPEG] ERROR 1203 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header header_length indicates ‘number’ bytes, ‘number’ decoded
The parser did not decode the number of bytes indicated by the header_length field. This could indicate a problem in the header_length field value, or an invalid stream_id.

>>> [MPEG] ERROR 1204 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header marker_bit ‘number’ is 0
All marker_bit fields should be encoded as ‘1’.

>>> [MPEG] ERROR 1205 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header marker 11 expected
The marker ‘11’, following the stream_id field was not found in the stream.

>>> [MPEG] ERROR 1210 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header rate_bound is ‘value’ Mbit/s, but mux_rate is ‘value’
The program_mux_rate from the Pack header in the PS is larger than the muxrate specified by the rate_bound field. The rate_bound field specifies the maximum combined muxrate of all ES in the PS.

>>> [MPEG] ERROR 1211 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header rate_bound is ‘value’ Mbit/s, previous occurrence ‘value’
Once specified, the rate_bound should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1212 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header audio_bound is ‘value’, should be <= 32
MPEG specifies a maximum of 32 audio ES in a PS.

>>> [MPEG] ERROR 1213 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header audio_bound is ‘value’, previous occurrence ‘value’
Once specified, the audio_bound should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1214 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header fixed_flag is ‘value’, was ‘value’ on previous occurrence
Once specified, the fixed_flag should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1215 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header CSPS_flag is ‘value’, was ‘value’ on previous occurrence
Once specified, the CSPS_flag should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1220 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header audio_lock_flag is ‘value’, was ‘value’ on previous occurrence
Once specified, the audio_lock_flag should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1230 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header video_lock_flag is ‘value’, was ‘value’ on previous occurrence
Once specified, the video_lock_flag should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1240 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header video_bound is ‘value’, should be <= 16
MPEG specifies a maximum of 16 video ES in a PS.

>>> [MPEG] ERROR 1241 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
System_header video_bound is ‘value’, previous occurrence ‘value’
Once specified, the video_bound should remain the same in all system_headers in the PS.

>>> [MPEG] ERROR 1244 (ref. MPEG-1 Systems 2.4.4.2) :
System_header reserved_byte is 'hexadecimal value', should be 0xFF
All reserved bytes should be coded as all-'1', i.e. 0xFF.

>>> [MPEG] ODDITY 1245 :
System_header empty STD-buffer list
No streams were specified at the end of the system_header, where normally the P-STD_buffer fields are defined. This would mean that no elementary streams are present in the PS, as each ES present in the PS shall have its P-STD_buffer_bound_scale and P-STD_buffer_size_bound specified exactly once in each system_header.

>>> [MPEG] ERROR 1246 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header stream_id is 'value', should be 0xB9, 0xB8, or in 0xBC..0xFF

>>> [MPEG] ERROR 1247 (ref. MPEG Systems 2.4.4.2 | Tab.2-19) :
System_header stream_id 'value' refers to a reserved (data)stream

>>> [MPEG] ERROR 1250 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
'Audio | Video' wildcard for STD_buffer_size_bound redefines streams 'numbers'
The system_header specified an Audio or Video wildcard stream_id, but an Audio or Video stream was already defined in the P-STD_buffer list.

>>> [MPEG] ERROR 1251 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
STD_buffer_size_bound for stream 'hexadecimal stream ID' missing, was previously explicitly defined
The definition of the stream mentioned in the error was not found in the current system_header, but was defined in the previous system_header.

>>> [MPEG] ERROR 1252 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
System_header STD_buffer_bound_scale is 'value' for 'Audio | Video' stream
The P-STD_buffer_bound_scale should have the value '0' when the previous stream_id in the list indicates an Audio stream. The P-STD_buffer_bound_scale should have the value '1' when the previous stream_id in the list indicates an Video stream.

>>> [MPEG] ERROR 1253 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
STD_buffer_size_bound for stream 'hexadecimal stream ID' is 'value' bytes, previously 'value'
Once specified, the P- STD_buffer_size_bound should remain the same for a specific stream_id, in all system_headers in the PS.

>>> [MPEG] ERROR 1254 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
STD_buffer_size_bound defined for 'Audio | Video' stream 'hexadecimal stream ID' , wildcard has previously been used
The P- STD_buffer_size_bound is specified for an Audio or Video stream, but in a previous system_header, the P-STD_buffer_size_bound for this stream was specified using the wildcard stream_id for that type (Audio or Video) of streams.

>>> [MPEG] ERROR 1255 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
STD_buffer_size_bound redefined for stream 'hexadecimal stream ID'
Each ES present in the PS shall have its P-STD_buffer_bound_scale and P-STD_buffer_size_bound specified exactly once in each system_header. This error reports the stream_id of a stream that has been defined more than once.

>>> [MPEG] ERROR 1256 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
STD_buffer_size_bound for stream 'hexadecimal stream ID' previously not defined
Once specified, the P- STD_buffer list should remain the same for all system_headers in the PS. This stream_id has previously not defined.

>>> [MPEG] ERROR 1257 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
STD_buffer_size_bound for 'Audio | Video' stream 'number' is 'value' bytes, previously 'value'

Once specified, the P- STD_buffer_size_bound specified with the wildcard stream_id should remain the same for all stream_ids, in all system_headers in the PS.

>>> [MPEG] ERROR 1260 (ref. MPEG Systems 2.4.5.6 | 2.7.8) :
Packet not preceded by system header in this stream

9.2.1.3 MPEG PES checks

>>> [MPEG] SYNTAX ERROR 1400 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet header too short
This error indicates a problem with the PES_packet_length field or with the decoding of the PES_packet, possibly due to some flags that are set to '1' inadvertently. Generally, the error is caused by the parser that wants to parse more data than available.

>>> [MPEG] SYNTAX ERROR 1401 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet marker_bit 'number' is 0
All marker_bits should be codes as '1'.

>>> [MPEG] SYNTAX ERROR 1402 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet has invalid timestamp mark ('value')
This error is reported when the marker bits just before the PTS[32..30] are invalid. These bits should be coded as '0010' when only the PTS is encoded, and '0011' when the PTS_DTS_flags indicate that both DTS and PTS are encoded.

>>> [MPEG] SYNTAX ERROR 1403 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet timestamp mark 0001 expected
This error is reported when the 4 bits following the PTS[14..0] and marker_bit fields are not coded as '0001', when the PTS_DTS_flags indicate that both PTS and DTS should be encoded.

>>> [MPEG] ERROR 1411 (ref. MPEG Systems 2.4.4.2 | Tab.2-19) :
PES_packet reserved stream ID : 'hexadecimal stream ID'

>>> [MPEG] ERROR 1412 (ref. MPEG Systems 2.4.4.2 | Tab.2-19) :
PES_packet reserved data stream ID : 'hexadecimal stream ID'

>>> [MPEG] ERROR 1413 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
'number' PES audio streams active at time 'time string', bound only 'value'
At some time in the PS, more audio streams were active than specified in the system_header.

>>> [MPEG] ERROR 1414 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
STD_buffer_size_bound for 'stream type' stream 'number' missing, packet(s) exist
Packets from the reported stream type and number were found in the PS, but the P-STD_buffer_size_bound for the reported streams was not specified in the system_header by using the wildcard stream_id.

>>> [MPEG] ERROR 1419 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
Packet for stream 'hexadecimal stream ID' appears, no corresponding STD_buffer_size_bound in system header
Packets from the reported stream_id were found in the PS, but the P-STD_buffer_size_bound for the reported stream_id was not specified in the system_header.

>>> [MPEG] SYNTAX ERROR 1420 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet header is 'value' bytes, packet_length + 6 only 'value'
The decoded PES_packet is longer that the PES_packet_length + 6 bytes. This could be caused by some flags that have inadvertently been set to '1', causing the parser to parse more data then intended.

>>> [MPEG] PIPE ERROR 1421 (ref. MPEG Systems 2.4.4.3 | 2.5.3.7) :
Skipped packet extends past end of bit pipe

>>> [MPEG] SYNTAX ERROR 1422 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :

PES_packet_length is 'value', should be <= 'value'

>>> [MPEG] SYNTAX ERROR 1424 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet contains too many stuffing bytes ('number' > max 'number')
The maximum number of stuffing bytes in a PES_packet:
MPEG-1 16
MPEG-2 32

>>> [MPEG] SYNTAX ERROR 1425 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet stuffing byte['index'] is 'hexadecimal value', should be 0xFF
All stuffing bytes should be coded as 0xFF.

>>> [MPEG] ERROR 1428 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
'number' PES video streams active at time 'time string', bound only 'number'
At some time in the PS, more video streams were active than specified in the system_header.

>>> [MPEG] ERROR 1430 (ref. MPEG Systems 2.4.5.5 | 2.7.7) :
No STD_buffer_size in first packet of 'Audio | Video' stream 'number'
The first packet of a Video or Audio stream should specify the P-STD_buffer_size in the PES_packet extension part.

>>> [MPEG] ERROR 1431 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet P-STD_buffer_scale is 'value' in 'Audio | Video' packet
The P-STD_buffer_bound_scale should have the value '0' when the preceding stream_id indicates an Audio stream. The P-STD_buffer_bound_scale should have the value '1' when the preceding stream_id indicates a Video stream.

>>> [MPEG] ERROR 1432 (ref. MPEG Systems 2.4.6 | 2.7.9) :
PES_packet P-STD_buffer_size is 'value' bytes, should be <= 'value' for a CSPS stream
This error is only reported for Video PES_packets and issued for each PES_packet. The P-STD_buffer_size specified an invalid value. Valid values are calculated according:

BS_{vbv} = vbv_buffer size
BS_{add} = MAX(6144, $R_{v\,max}$ * 0.001)

$R_{v\,max}$ is the maximum video bitrate of the ES.

P-STD_buffer_size for video <= BSvbv + BSadd

>>> [MPEG] ERROR 1433 (ref. MPEG Systems 2.4.6 | 2.7.9) :
PES_packet P-STD buffer size is 'value' bytes, should be <= 4096 (CSPS)
This error is only reported for Audio PES_packets.

>>> [MPEG] ERROR 1434 (ref. MPEG Systems 2.4.4.2 | 2.5.3.6) :
PES_packet P-STD-buffer size is 'value' bytes > system_header P-STD buffer size bound ('value' bytes)
The P-STD_buffer_size from the PES_packet_header in the Packet is larger than the P-STD_buffer_size specified by the system header.

>>> [MPEG] ODDITY 1435 :
PES_packet P-STD buffer size 0 specified !
PES_packet with the size '0' can only be used as some form of stuffing.

>>> [MPEG] ERROR 1437 (ref. MPEG Systems 2.4.6 | 2.7.9) :
PES P-STD_buffer_size is 'value' bytes, should be <= 'value' for a CSPS video stream
with bitrate 'value' bits/sec and a VBV buffer of 'value' bytes.
This error is only reported for Video PES_packets and only issued when a sequence header is found. The P-STD_buffer_size specified an invalid value. Valid values are calculated according:

BS_{vbv} = vbv_buffer size
BS_{add} = MAX(6144, $R_{v\ max}$ * 0.001)

$R_{v\ max}$ is the maximum video bitrate of the ES.

P-STD_buffer_size for video <= BSvbv + BSadd.

>>> [MPEG] ERROR 1440 (ref. MPEG Systems 2.4.5.4 | 2.7.5) :
'packet type' PES_Packet 'number' contains timestamp(s), no AU commences in it
A PTS may only be present in a Video or Audio elementary stream PES packet header if the first byte of a picture start code (Video ES) or the first byte of an Audio access unit is contained in the PES_packet.

>>> [MPEG] ERROR 1442 (ref. MPEG Systems 2.4.5.4 | 2.4.3.7) :
This Audio PES_packet has a decoding timestamp
A PES_packet header containing Audio elementary data is not allowed to contain a DTS.

>>> [MPEG] ERROR 1443 (ref. MPEG Systems 2.4.5.4 | 2.7.5) :
Video PES_Packet 'number' contains 'no|no PTS, but contains a' DTS
This error is reported in these two instances:
- The PES_header specifies a DTS, but the ES contains MPEG-1 Video and the picture that starts in the current PES_packet has a picture_type 'D'.
- The PES_header specified a DTS, but not a PTS.

>>> [MPEG] ERROR 1444 (ref. MPEG Systems 2.4.5.4 | 2.7.5) :
'DTS|PTS' of 'packet type' PES_Packet 'number' is 'time string', should be 'time string'
This error is reported when the decoding_time for the next AU does not correspond to the PTS or DTS of the PES_packet.

>>> [MPEG] ERROR 1445 (ref. MPEG Systems 2.4.5.4 | 2.7.5) :
'picture position' I/P picture's PTS - DTS offset is 'value', should be 'value'
Checks if the decoding time of the first, current, or previous I/P picture is correct.

>>> [MPEG] ERROR 1446 (ref. MPEG Systems 2.4.5.4 | 2.7.5) :
PTS and DTS are identical ('time string'), should be different
The PTS and the DTS cannot be equal. This would mean that the decoding of the AU would happen instantly.

>>> [MPEG] ERROR 1447 (ref. MPEG Systems 2.4.5.3 | 2.7.4) :
'packet type' PES_Packet PTS difference with previous PTS is 'time string',
should be <= 0.7 seconds or 63000 ticks

>>> [MPEG] ERROR 1448 (ref. MPEG Systems 2.4.5.1 | 2.5.2.3) :
PES STD buffer underflow at decoding time 'time string' [msec] of AU 'number' ('number' bytes)
This error reports that, according to the STD buffer calculations, one of these errors occurred:
- The AU due for removal is larger than the remaining STD buffer contents
- The size of the AU due for removal is not yet known (because the end of the AU had not been received in the stream)

>>> [MPEG] ERROR 1449 (ref. MPEG Systems 2.4.5.1 | 2.5.2.3) :
PES STD buffer overflow at decoding time 'time string' [msec] of AU 'number' ('number' bytes, buffer size 'value')
This error reports that, according to the STD buffer calculations, the payload of the PES_packet added to the STD buffer contents will cause an STD buffer contents larger than the allowed STD buffer size. The error reports this fact, but the PES_packet payload is still being added to the STD buffer contents. This error could be due to erroneous PTS/DTS values or it could be an encoding problem.

>>> [MPEG] ERROR 1450 (ref. MPEG Systems 2.4.5.1 | 2.5.2.3) :

First byte of AU in 'packet type' PES_packet 'number' arrives at 'time string' or 'number' ticks after its decoding time 'time string'.

The decoding time for the AU mentioned in the error has already past, but the first byte was just received. This could result in some decoding problems and associated Audio or visual errors.

>>> [MPEG] ERROR 1451 (ref. MPEG Systems 2.4.5.1 | 2.5.2.3) :

First byte of AU in 'packet type' PES_packet 'number' arrives at 'time string' or 'time string' before its decoding time 'time string', which is more than the allowed 'number' second ('number' ticks)

Data from any AU is only allowed to remain in the STD buffer for a limited period. The first byte of the AU mentioned in this error was received more than that limited period before it should be decoded, meaning data would have to stay in the STD buffer for too long. The period is shown in the following table:

AU	time (sec)
still picture	60
other	1

>>> [MPEG] ERROR 1460 (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :

PES_packet padding byte 'number' is 'hexadecimal value', should be 0xFF

All padding bytes should be coded as '0xFF'.

9.2.1.4 MPEG Sequence header checks

>>> [MPEG] ERROR 1500 :

Video sequence ('number') in 'stream type' 'hexadecimal stream ID' is not properly terminated, e.g. missing sequence_end_code.

No sequence_end_code found before the end of the video ES.

>>> [MPEG] SYNTAX ERROR 1502 (ref. MPEG Video 2.4.2.3 | 6.2.2) :

'code type string' expected

This error is reported when the parser expect one of these codes:

- Sequence header start code
- Extension header start code
- User_data start code
- Group of Pictures header start code
- Picture header start code
- Sequence_end_code

>>> [MPEG] SYNTAX ERROR 1503 (ref. MPEG Video 2.4.3.2 | 6.3.3) :

Sequence_header marker bit is 0

All marker_bits in the sequence_header should be coded as '1'.

>>> [MPEG] SYNTAX ERROR 1504 (ref. MPEG Video 2.4.2.3 | 6.2.2) :

Sequence_header_code expected

After parsing the last token of the previous picture, a new Sequence_header must be decoded in the Video ES, but the Sequence_header start code was not found in the PES_packet payload following the last token of the previous picture.

>>> [MPEG] SYNTAX ERROR 1505 (ref. MPEG Video 2.4.2.3 | 6.2.2) :

Sequence_header extension, user_data, group_start_code (or picture_start_code, MPEG-2 only) expected

At this point in the parsing process, only these start codes are allowed.

>>> [MPEG] SYNTAX ERROR 1506 (ref. MPEG Video 2.4.2.3 | 6.2.2) :

Sequence_header user_data or group_start_code expected

At this point in the parsing process, only these start codes are allowed.

>>> [MPEG] SYNTAX ERROR 1507 (ref. MPEG Video 2.4.2.3 | 6.2.2) :

Expecting group/sequence_header/sequence_end start code

At this point in the parsing process, only these start codes are allowed.

>>> [MPEG] ERROR 1510 (ref. MPEG Video ? | 6.3.3) :
Sequence_header horizontal_size is 0
A horizontal_size of '0' is not allowed, a picture cannot be '0' pixels wide. The horizontal_size should be larger than '0'.

>>> [MPEG] ERROR 1511 (ref. MPEG-1 Video 2.4.3.2) :
Sequence_header horizontal_size is 'value', should be <= 768
when the constrained_parameters_flag is set.
In a constrained parameter stream, the horizontal_size should be less than, or equal to 768.

>>> [MPEG] ERROR 1513 (ref. MPEG Video 2.4.1 | 6.1.1.6) :
Sequence_header horizontal_size is 'value', previously defined to be 'value'
All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1514 (ref. MPEG Video ? | 6.3.3) :
Sequence_header vertical_size is 0
A vertical_size of '0' is not allowed, a picture cannot be '0' pixels high. The vertical_size should be larger than '0'.

>>> [MPEG] ERROR 1515 (ref. MPEG-1 Video 2.4.3.2) :
Sequence_header vertical_size is 'value', should be <= 576
when the constrained_parameters_flag is set.
In a constrained parameter stream, the vertical_size should be less than, or equal to 576.

>>> [MPEG] ERROR 1517 (ref. MPEG Video 2.4.1 | 6.1.1.6) :
Sequence_header vertical_size is 'value', previously defined to be 'value'
All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1518 (ref. MPEG-1 Video 2.4.3.2) :
Picture area is 'value' macroblocks, should be <= 396
when the constrained_parameters_flag is set.
In a constrained parameter stream, the picture area should be less than, or equal to 396.

>>> [MPEG] ERROR 1521 (ref. MPEG-1 Video 2.4.3.2) :
Product of picture area, and picture rate, is 'value', should be <= 9900
when the constrained_parameters_flag is set.
In a constrained parameter stream, the product of the picture area and the picture rate should be less than, or equal to 396 * 25 (equals 9900).

>>> [MPEG] ERROR 1530 (ref. MPEG Video 2.4.3.2 | Tab.6-3) :
Sequence_header aspect_ratio field has forbidden value 0
The aspect_ratio value '0000b' is a forbidden value and shall not be used

>>> [MPEG] ERROR 1531 (ref. MPEG Video 2.4.3.2 | Tab.6-3 and Tab. 8-5) :
Sequence_header aspect_ratio has reserved value 'value', should be within range 'minimum value'...'maximum value'
The aspect_ratio value may only specify a value in the reported range. This range is specified as:

	minimum value	**maximum value**
MPEG-1	1	14 (0xE)
MPEG-2	1	4

>>> [MPEG] ERROR 1532 (ref. MPEG Video 2.4.1 | 6.1.1.6) :
Sequence_header pixel aspect ratio is 'value', previously defined to be 'value'

All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1533 (ref. MPEG Video 2.4.3.2 | Tab.6-4) :
Sequence_header : illegal frame_rate_code 'value'
The frame_rate_code must specify a value between '0001b' and '1000b'.

>>> [MPEG] ERROR 1534 (ref. MPEG-1 Video 2.4.3.2) :
Sequence_header frame_rate is 'value' Hz, should be <= 30
when the constrained_parameters_flag is set.
In a constrained parameter stream, the frame_rate should be less than, or equal to 30.

>>> [MPEG] ERROR 1536 (ref. MPEG Video 2.4.1 | 6.1.1.6) :
Sequence_header frame_rate is 'value', previously defined to be 'value'
All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1537 (ref. MPEG Video 2.4.3.2 | 6.3.3) :
Sequence_header bit_rate is 0
The bit_rate field should specify a bitrate of more than 0 bit/sec.

>>> [MPEG] ERROR 1542 (ref. MPEG-1 Video 2.4.3.2) :
Sequence_header bit_rate is 'value', should be fixed and <= 1.856 Mbit/s
when the constrained_parameters_flag is set.
In a constrained parameter stream, the bit_rate should be fixed and less than, or equal to 1.836 Mbit/sec.

>>> [MPEG] ERROR 1543 (ref. MPEG Video 2.4.1 | 6.1.1.6) :
Sequence_header bit_rate is 'value', previously defined to be 'value'
All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1544 (ref. MPEG-1 Video 2.4.3.2) :
Sequence_header buffer size is 'value' Kbytes, should be <= 40
when the constrained_parameters_flag is set.
In a constrained parameter stream, the buffer size should be less than, or equal to 40 Kbytes.

>>> [MPEG] ERROR 1547 (ref. MPEG Video 2.4.1 | 6.1.1.6) :
Sequence_header buffer size is 'value' Kbytes, previously defined to be 'value' Kbytes
All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1548 (ref. MPEG-1 Video 2.4.1) :
Sequence_header constrained_parameters_flag is 'value', previously defined to be 'value'
All of the fields in repeated sequence_headers shall have the same values as the first sequence_header in the stream, excluding the fields defining the quantisation matrices.

>>> [MPEG] ERROR 1550 (ref. MPEG Video 2.4.3.2 | 6.3.11) :
Sequence_header intra_quantiser_matrix['index'] is 0.
The intra_quantiser_matrix value '0' is forbidden and must be in range [1..255].

>>> [MPEG] ERROR 1551 (ref. MPEG Video 2.4.3.2 | 6.3.11) :
Sequence_header intra_quantiser_matrix[0] is 'value', should be 8
The first intra_quantiser_matrix should always be '8'.

>>> [MPEG] ERROR 1552 (ref. MPEG Video 2.4.3.2 | 6.3.11) :
Sequence_header non_intra_quantiser_matrix['index'] is 0
The non_intra_quantiser_matrix shall not specify the value '0'.

>>> [MPEG] ERROR 1553 (ref. MPEG Video 2.4.3.2 | 6.3.4.1) :
Sequence_header's user_data contains ‘number’ consecutive zero bits
The user_data contained enough consecutive ‘0’ bits that an emulation of a start code could occur. The user_data is not allowed to emulate a start code, therefore the user_data shall be coded in such a way that more than 23 consecutive ‘0’ bits do not occur.

>>> [MPEG] ERROR 1599 (ref. MPEG-1 Video 2.4.3.2) :
Sequence_header contains ‘number’ bytes of sequence_extension_data

9.2.1.5 MPEG GOP checks

>>> [MPEG] SYNTAX ERROR 1621 (ref. MPEG Video 2.4.2.4 | 6.2.2.6) :
GOP next_start_code() failed
The parser could not find a valid GOP start code in the stream at the position reported. This could indicate a problem in the preceding GOP. The parser will recover to the next valid start code.

>>> [MPEG] ERROR 1622 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP marker_bit in time_code is 0
All marker_bits should be coded as ‘1’.

>>> [MPEG] SYNTAX ERROR 1623 (ref. MPEG Video 2.4.2.4 | 6.2.2.6) :
GOP extension, user_data or picture_start_code expected
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, only the reported start codes may occur. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1624 (ref. MPEG Video 2.4.2.4 | 6.2.2.6) :
GOP user_data or picture_start_code expected
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, only the reported start codes may occur. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1625 (ref. MPEG Video 2.4.2.4 | 6.2.2.6) :
Expecting picture/group/sequence_header/sequence_end start code
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, one of the reported start codes must occur. The parser will recover to the next valid start code.

>>> [MPEG] ERROR 1630 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP drop_frame_flag is 1, picture rate is ‘value’
The drop_frame_flag may only be set to ‘1’ when the frame rate specified 29.97 Hz.

>>> [MPEG] ERROR 1631 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP time_code_hours is ‘value’, should be in 0..23

>>> [MPEG] ERROR 1632 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP time_code_minutes is ‘value’, should be in 0..59

>>> [MPEG] ERROR 1633 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP time_code_seconds is ‘value’, should be in 0..59

>>> [MPEG] ERROR 1634 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP time_code_pictures is ‘value’, should be in ‘minimum value’..’maximum value’
The GOP time_code_pictures must specify a value within the reported range. The range is limited at the upper end by the frame_rate, rounded to the nearest integral number of pictures per second. The lower limit is normally ‘0’, but when the GOP specified the drop_frame_flag value ‘1’ AND the time_code_minutes does not equal ‘0’, ‘10’, ‘20’, ‘30’, ‘40’, or ‘50’, the minimum value is ‘2’.

>>> [MPEG] ODDITY 1635 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP time_code is ‘timecode string’, expected ‘timecode string’

The GOP timecode should contain no gaps, but the calculated GOP time_code differs from the encoded time_code.

>>> [MPEG] ODDITY 1636 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP closed_gop and broken_link both 1
The closed_gop flag is used to indicate the nature of the predictions used in the first consecutive B-pictures immediately following the first coded I-frame following the GOP header. A value of '1' indicates that these B-pictures are encoded using only backward prediction or intra coding. This cannot coincide with the broken_link flag, which is used to indicate that the first consecutive B-pictures immediately following the first coded I-frame following the GOP header may not be correctly decoded because the reference frame which is used for prediction is not available. Thus B-pictures with the closed_gop flag set to '1' can always be correctly decoded, negating the broken_link flag.

>>> [MPEG] ERROR 1640 (ref. MPEG Video 2.4.3.3 | 6.3.1) :
GOP contains 'number' bytes of group_extension_data
However extension_data is not allowed following a GOP_header.

>>> [MPEG] ERROR 1641 (ref. MPEG Video 2.4.3.3 | 6.3.4.1) :
GOP's user_data contains 'number' consecutive zero bits
The user_data contained enough consecutive '0' bits that an emulation of a start code could occur. The user_data is not allowed to emulate a start code, therefore the user_data shall be coded in such a way that more than 23 consecutive '0' bits do not occur.

>>> [MPEG] ERROR 1642 (ref. MPEG Video 2.4.1 | Compl 9.2.1.3) :
GOP ends with too few B-pictures
There are less B-pictures than expected in the current GOP.

9.2.1.6 MPEG Picture checks

>>> [MPEG] SYNTAX ERROR 1650 (ref. MPEG Video 2.4.2.5 | 6.2.3) :
Picture next_start_code() failed
The parser could not find a valid picture start code in the stream at the position reported. This could indicate a problem in the preceding picture. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1653 (ref. MPEG Video 2.4.2.5 | 6.2.3) :
Picture extension, user_data or slice_start_code expected
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, only the reported start codes may occur. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1654 (ref. MPEG Video 2.4.2.5 | 6.2.3) :
Picture user_data or slice_start_code expected
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, only the reported start codes may occur. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1655 (ref. MPEG Video 2.4.2.5 | 6.2.3) :
Picture slice_start_code expected
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, only the reported start codes may occur. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1656 (ref. MPEG Video 2.4.2.5 | 6.2.3) :
Expecting slice/picture/group/sequence_header/sequence_end start code
The parser could not find any of the mentioned start codes in the stream at the position reported. At this position, one of the reported start codes must occur. The parser will recover to the next valid start code.

>>> [MPEG] ERROR 1661 (ref. MPEG Video 2.4.3.4 | 6.3.9) :
Picture temporal_reference is 'value', should be 'number'
This error is reported in these three distinct cases:

- The temporal_reference of the reported picture must be the same as the previous, because the picture is coded as two field pictures.
- The temporal_reference of the reported picture must be incremented by 1 modulo 1024 over the previous picture
- The temporal_reference of the reported picture must be '0', because it follows a GOP header

>>> [MPEG] ERROR 1662 (ref. MPEG Video 2.4.3.4 | 6.3.9) :
Picture invalid picture_coding_type 'number' ('picture type string')
The picture type should describe a value as stated in this table:

	minimum range	**maximum range**
MPEG1	001 (I)	100 (D)
MPEG2	001 (I)	011 (B)

>>> [MPEG] ERROR 1663 (ref. MPEG-1 Video 2.4.3.4) :
Combination D- and non-D pictures in same sequence

>>> [MPEG] ERROR 1666 (ref. MPEG Video 2.4.1 | 6.1.1.7) :
First picture in GOP has type 'picture type string', should be an I-picture

>>> [MPEG] ERROR 1667 (ref. MPEG Video 2.4.3.4 | 6.3.9) :
Picture has type 'picture type string', temporal reference of picture 'value' indicates it should be B
This error is generated when a picture's encoded temporal reference value does not match the coding sequence order, e.g. after coding an I-P-picture pair, the number of B-pictures that should follow can be derived from the temporal reference value difference of the I- and P-picture. If there are more or less B-pictures encoded, this error is generated.

>>> [MPEG] ERROR 1670 (ref. MPEG-1 Video 2.4.3.4) :
Picture vbv_delay is 'value', should be 0xFFFF (variable bit_rate)
In a variable bit rate Video stream, all vbv_delay values should be encoded with the value '0xFFFF'.

>>> [MPEG] ERROR 1671 (ref. MPEG-1 Video 2.4.3.4) :
Picture forward_f_code is 0
The forward_f_code field specified the reserved value '0'.

>>> [MPEG] ERROR 1672 (ref. MPEG-1 Video 2.4.3.2) :
Picture forward_f_code 'value' exceeds '4'
when the constrained_parameters_flag is set

>>> [MPEG] ERROR 1674 (ref. MPEG-1 Video 2.4.3.4) :
Picture backward_f_code is 0
The backward_f_code field specified the reserved value '0'.

>>> [MPEG] ERROR 1675 (ref. MPEG-1 Video 2.4.3.2) :
Picture backward_f_code 'value' exceeds '4'
when the constrained_parameters_flag is set

>>> [MPEG] ERROR 1680 (ref. MPEG Video 2.4.3.4 | 6.3.9) :
Picture contains 'number' bytes of extra_information_picture data
No extra_information_picture data may be encoded in the Picture header

>>> [MPEG] ERROR 1681 (ref. MPEG-1 Video 2.4.3.4) :
Picture contains 'number' bytes of group_extension_data
No group_extension_data may be encoded in the Picture header

>>> [MPEG] ERROR 1682 (ref. MPEG Video 2.4.3.4 | 6.3.4.1) :
Picture's user_data contains ‘number’ consecutive zero bits
The user_data contained enough consecutive ‘0’ bits that an emulation of a start code could occur. The user_data is not allowed to emulate a start code, therefore the user_data shall be coded in such a way that more than 23 consecutive ‘0’ bits do not occur.

>>> [MPEG] ERROR 1690 (ref. MPEG Video 2-C.1 | Ann.C) :
VBV buffer underflow for picture ‘number’ (‘number’ bytes, m=’number’)
The VBV buffer model detected an underflow in the VBV buffer model.

>>> [MPEG] ERROR 1691 (ref. MPEG Video 2-C.1 | Ann.C) :
VBV buffer overflow for picture ‘number’ (‘number’ bytes, m=’number’)
The VBV buffer model detected an overflow in the VBV buffer model.

>>> [MPEG] ERROR 1692 (ref. MPEG-1 Video 2.4.3.4) :
vbv_delay for picture ‘number’ is ‘value’, should be in range ‘minimum value’ .. ‘maximum value’

>>> [MPEG] ERROR 1695 (ref. MPEG Video C.3.1) :
Decoding time of picture ‘number’, is before it has been received completely.
The end of the reported picture was not found before the decoding time. This would mean that the decoder will be unable to decode the picture completely and display it without distortions.

>>> [MPEG] ERROR 1696 (ref. MPEG Video C.3.1) :
Vbv_delay value ‘value’ incorrect: leads to negative or zero time to receive all bits of previous picture

>>> [MPEG] ERROR 1697 (ref. MPEG Video C.3.1) :
Actual bit_rate ‘value’ Mbit/s exceeds maximum bit_rate ‘maximum value’ Mbit/s
as specified in sequence header and sequence extension

9.2.1.7 MPEG Slice checks

>>> [MPEG] ERROR 1751 (ref. MPEG Video 2.4.3.5 | 6.3.16) :
Slice vertical position ‘value’ exceeds picture height
The vertical position of the slice must specify a position within the picture boundaries.

>>> [MPEG] ERROR 1752 (ref. MPEG Video 2.4.3.5 | 6.3.16) :
Slice quantizer_scale_code has the forbidden value 0

>>> [MPEG] ERROR 1753 (ref. MPEG Video 2.4.3.5 | 6.3.16) :
Slice contains extra_information_slice data
No extra_information_slice may be encoded in the Slice header.

>>> [MPEG] SYNTAX ERROR 1754 (ref. MPEG Video 2.4.2.6 | 6.2.4) :
Expecting slice-start-code etc.
The parser could not find the mentioned start code in the stream at the position reported. At this position, the reported start code must occur. The parser will recover to the next valid start code.

>>> [MPEG] SYNTAX ERROR 1755 (ref. MPEG Video 2.4.2.6 | 6.2.4) :
Illegal next_start_code()
The parser did not find any valid start code in the stream, this error is normally preceded by SYNTAX ERROR 1754.

9.2.1.8 MPEG Macroblock checks

>>> [MPEG] SYNTAX ERROR 1770 (ref. MPEG-1 Video 2.4.3.6) :

end_of_macroblock field is 0

>>> [MPEG] SYNTAX ERROR 1771 (ref. MPEG Video 2-B.1 | B.1) :
Invalid macroblock_address_increment

>>> [MPEG] ERROR 1772 (ref. MPEG Video 2.4.4.5 | Compl. 9.2.1.5) :
Macroblock 'number' not intra-coded in 132 P-pictures
Each macroblock should be intra-coded at least once per every 132 times it is coded in a P-picture without intervening I-picture.

>>> [MPEG] ERROR 1773 (ref. MPEG Video 2.4.1 | 6.3.17) :
Macroblock number 'number' outside picture
The macroblock absolute position derived from the macroblock_address falls outside the specified picture dimensions.

>>> [MPEG] ERROR 1774 (ref. MPEG Video 2.4.3.5 | 6.3.17) :
Macroblock address increment 'value' exceeds picture width
The macroblock absolute position derived from the macroblock_address falls outside the specified picture dimensions.

>>> [MPEG] ERROR 1775 (ref. MPEG Video 2.4.1 | 6.3.17) :
First macroblock in picture has number 'value'.
The first macroblock should not be skipped and must have number '1'.

>>> [MPEG] ERROR 1776 (ref. MPEG Video 2.4.1 | 6.3.17) :
Inter-slice gap between macroblock 'number' and 'number'
Every slice should be encoded.

>>> [MPEG] ERROR 1777 (ref. MPEG Video 2.4.4.4 | 6.3.17) :
Macroblock(s) skipped between 'number' (intra) and 'number'
In a B-picture there shall be no skipped macroblocks immediately following an intra_coded macroblock.

>>> [MPEG] SYNTAX ERROR 1778 (ref. MPEG Video 2-B.2 | B.2) :
Invalid macroblock_type

>>> [MPEG] ERROR 1779 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP closed, but macroblock refers to previous GOP
In a closed GOP (closed_gop == 1) no backward prediction (referencing previous picture data) is allowed for the B-pictures immediately following the first I-picture in the GOP.

>>> [MPEG] ERROR 1780 (ref. MPEG Video 2.4.3.3 | 6.3.8) :
GOP first in sequence, broken_link not set, but macroblock refers to previous GOP
In a closed GOP (closed_gop == 1) no backward prediction (referencing previous picture data) is allowed for the B-pictures immediately following the first I-picture in the GOP.

>>> [MPEG] ERROR 1781 (ref. MPEG Video 2.4.3.6 | Tab.7-6) :
Macroblock quantizer_scale_code is 0

>>> [MPEG] SYNTAX ERROR 1782 (ref. MPEG Video 2-B.4 | B.4) :
Macroblock illegal 'error string'
This error is reported for illegal use of:

- motion_horizontal_forward_code
- motion_vertical_forward_code
- motion_horizontal_backward_code
- motion_vertical_backward_code
- motion_code
- dmvector

>>> [MPEG] ERROR 1783 (ref. MPEG Video 2.4.4.2 | 7.6.3) :
Reconstruction of 'vertical|horizontal' component of 'first|second|first dual prime|second dual prime' 'forward|backward' vector for macroblock 'number' failed: reconstructed vector component out of allowable motion vector range
The allowable motion vector range is described in Table 7-8.

>>> [MPEG] ERROR 1784 (ref. MPEG Video 2.4.4.2 | 7.6.3.8) :
Horizontal component 'number' of 'first|second|first dual prime|second dual prime' 'forward|backward' vector for macroblock 'number' out of picture boundaries

>>> [MPEG] SYNTAX ERROR 1785 (ref. MPEG Video 2-B.3 | B.3) :
Macroblock illegal coded_block_pattern

>>> [MPEG] ERROR 1786 (ref. MPEG Video 2.4.4.2 | 7.6.3.8) :
Vertical component 'number' of 'first|second|first dual prime|second dual prime' 'forward|backward' vector for macroblock 'number' out of picture boundaries

9.2.1.9 MPEG Block checks

>>> [MPEG] SYNTAX ERROR 1801 (ref. MPEG Video 2-B.5 | B.5) :
Block invalid 'dct_dc_size_luminance | dct_dc_size_chrominance'

>>> [MPEG] SYNTAX ERROR 1802 (ref. MPEG Video 2-B.5 | B.5) :
Block invalid 'dct_coeff_first | dct_coeff_next'
VLC decoding fails for the indicated DCT coefficient.

>>> [MPEG] ERROR 1803 (ref. MPEG Video 2.4.3.7 | 7.2) :
Block index is 'number'
Decoding error occurred causing the block index value to exceed 63.

>>> [MPEG] ERROR 1804 (ref. MPEG Video 2-B.5 | B.5) :
Block invalid DCT escape code

9.2.1.10 MPEG Audio checks

>>> [MPEG] SYNTAX ERROR 1851 :
Audio sequence does not start with a syncword
This error is generated for the first audio frame, in two distinct cases:
- The audio parser finished parsing the base frame, but the next bytes in the stream are not the extension syncword.
- The audio parser is about to start parsing the base frame, but the next bytes in the stream are not the baseframe syncword.

>>> [MPEG] SYNTAX ERROR 1852 :
No syncword where frame length indicates it should be
This error is generated, in two distinct cases:
- The audio parser finished parsing the base frame, but the next bytes in the stream are not the extension syncword.
- The audio parser is about to start parsing the base frame, but the next bytes in the stream are not the baseframe syncword.

This error is basically identical to ERROR 1851, with the exception that this error is generated when the check fails for an audio frame other than the first in the PES.

>>> [MPEG] INFORMATION 1860 :
Recovering ... 'number' bytes skipped.

The audio parser could not parse the previous audio frame(s) correctly and skipped the number of bytes indicated to a valid syncword. Normal parsing is resumed.

>>> [MPEG] ERROR 1870 :
Audio frame not complete
More base frames were found in the ES than there were extension frames, making the last baseframe incomplete.

>>> [MPEG] SYNTAX ERROR 1871 :
Amount of audio information exceeds frame length
This error reports that the amount of data the in the frame does not correspond to the frame_length field. This usually due to the audio parser assuming MPEG-2, while the stream is encoded as MPEG-1. The bytes following the base frame are interpreted as the multi-channel extension. This error can also occur due to erroneous flags in the stream.

>>> [MPEG] SYNTAX ERROR 1891 (ref. MPEG Audio 2.4.2.2) :
Audio frame ID is 0
The Audio ID field should be coded as '1'.

>>> [MPEG] ERROR 1892 (ref. MPEG Audio 2.4.2.2) :
Audio frame layer invalid
The frame layer should not describe the reserved value '0'.

>>> [MPEG] ERROR 1895 (ref. MPEG Audio 2.4.2.2) :
Audio : Forbidden bitrate
The bitrate field must be larger than the value '0xF'.

>>> [MPEG] ERROR 1896 (ref. MPEG Audio 2.4.2.2) :
Audio : Combination layer II, bitrate 'value' Kbit/s and 'mode type string' mode
For layer II MPEG audio, these bitrate/mode combinations are allowed:

Mode	**allowed bitrates**				
Single channel	32	48	56	80	Kbit/s
>Single channel	224	256	320	384	Kbit/s

>>> [MPEG] ERROR 1899 (ref. MPEG Audio 2.4.2.2) :
Audio : Invalid sampling_frequency
The sampling_frequency should not describe the reserved value '0x3'.

>>> [MPEG] ERROR 1901 (ref. MPEG Audio 2.4.2.2) :
Padding bit should have been 'value'

>>> [MPEG] ERROR 1903 (ref. MPEG Audio 2.4.2.2) :
Audio : Invalid emphasis
The emphasis should not describe the reserved value '0x2'.

>>> [MPEG] ERROR 1910 (ref. MPEG Audio 2.4.3.1 & MPEG-2 Audio 2.5.2.10 & 2.5.2.14) :
CRC error in 'frame type string' audio data: calculated value 'hexadecimal CRC value', value in 'frame type string' 'hexadecimal CRC value'
This error is generated when the calculated CRC does not correspond with the CRC found encoded in the stream for these locations of the CRC:

- Multichannel CRC
- Base frame CRC
- Extension frame CRC
- Application specific CRC

>>> [MPEG] ERROR 1911 (ref. MPEG Audio 3-B.1) :
Scalefactor'index' refers to index 63

A scalefactor specified a reserved value ‘63’. Together with the type of the scalefactor is also reported the index, which indicates the channel, sub-band and scalefactor number, depending on the scalefactor type.

>>> [MPEG] ERROR 1912 (ref. MPEG-2 Compliance 2.5.2.1.2 & 2.5.2.1.3 & 2.5.2.2.2) :
Sample’index’ value ‘value’ outside valid range [0, ‘maximum value’>
The reported sample or sample_code value is not within the valid range. The range is reported by the error, with the upper limit not included in the range of valid values. Together with the type of the sample is also reported the index, which indicates the channel, group and sub-band, depending on the sample type.

>>> [MPEG] ERROR 1913 (ref. MPEG-2 Compliance 2.5.2.1.2) :
allocation’index’ illegal value 0xF
An allocation specified a reserved value ‘0xF’. Also reported is the index, which indicates the channel and sub-band of the allocation.

9.2.2 MPEG-2 Checks

9.2.2.1 MPEG-2 PS checks

>>> [MPEG-2] ERROR 2301 (ref. MPEG-2 Systems various) :
Marker_bit is 0
All marker bits should be coded as '1'.

>>> [MPEG-2] SYNTAX ERROR 2308 (ref. MPEG-2 Systems 2.5.3.4) :
Pack_header marker 01 expected

>>> [MPEG-2] SYNTAX ERROR 2310 (ref. MPEG-2 Systems 2.5.3.4) :
Pack : specified stuffing length 'value' exceeds the pack header size
The stuffing should fit within the Pack header. In this case, the stuffing length exceeds the pack header, which could be caused by erroneous flags in the pack header, causing the parser to parse too much data.

>>> [MPEG-2] ERROR 2311 (ref. MPEG-2 Systems 2.5.3.4) :
Pack stuffing length should be < 7, is 'value'
The minimum length of stuffing in a Pack is 7 bytes.

>>> [MPEG-2] SYNTAX ERROR 2312 (ref. MPEG-2 Systems 2.5.3.4) :
Pack stuffing byte 'number' should be 0xFF, is 'hexadecimal value'
All stuffing bytes should be coded as '0xFF'. This error could point out that the bytes currently being parsed as stuffing, could mean something different altogether, i.e. meaningful data.

>>> [MPEG-2] ERROR 2331 (ref. MPEG-2 Systems 2.5.3.6) :
System_header reserved_byte is 'hexadecimal value', should be 0x7F

9.2.2.2 MPEG-2 PES checks

>>> [MPEG-2] ERROR 2401 (ref. MPEG-2 Systems various) :
Marker_bit is 0, should be 1
All marker bits should be coded as '1'.

>>> [MPEG-2] INFORMATION 2410 :
Gap in 'packet type string' sequence of 'number' seconds.
This error reports a gap in the timestamps for the reported packet type. This error is generated at the start of:

- Picture header start
- Audio frame start
- Audio extension frame start
- Private-1 access unit start

The timestamp of the previous start of an AU increased with the duration of that AU must equal the timestamp of the AU that just started.

>>> [MPEG-2] SYNTAX ERROR 2420 (ref. MPEG-2 Systems 2.1.47) :
PES_packet 'number' reserved bits field is 'hexadecimal value'; all bits should be 1

>>> [MPEG-2] SYNTAX ERROR 2421 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet starts with an illegal start_code_prefix : 'hexadecimal value'
A start_code_prefix should always be '0x000001'.

>>> [MPEG-2] ERROR 2422 (ref. MPEG-2 Systems Tab.2-18) :
PES_packet stream_id 0xF9 illegal for a Program Stream
The use of stream_id 0xF9 (Ancillary stream) is not allowed in a Transport stream.

>>> [MPEG-2] ERROR 2423 (ref. MPEG-2 Systems Tab.2-18) :
PES_packet stream_id 'hexadecimal stream id' illegal for a Transport Stream
The use of stream_id 0xBC (Program stream map) and 0xFF (Program stream directory) is not allowed in a Transport stream.

>>> [MPEG-2] ERROR 2430 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet length 0 only allowed in Transport Streams

>>> [MPEG-2] ERROR 2431 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet length 0 only allowed in TS video streams (actual ID = 'hexadecimal stream id')
Only the PES_packet with a stream_id belonging to a video stream in a Transport stream, may specify a PES_packet length of 0 bytes.

>>> [MPEG-2] SYNTAX ERROR 2432 (ref. MPEG-2 Systems 2.4.3.6) :
PES_packet 2-bit marker is not '10'

>>> [MPEG-2] ODDITY 2433 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet data_alignment_indicator set in non-audio or -video stream
The data_alignment_indicator can only be used in an Audio or Video stream. This error reports the data_alignment_indicator is set to '1' in any other stream.

>>> [MPEG-2] ERROR 2434 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet data_alignment_indicator set : no valid PES_packet data alignment
The data_alignment_indicator is used to indicate that the PES_header is immediately followed by a video start code or audio syncword.

>>> [MPEG-2] ERROR 2435 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet data_alignment_indicator set : data alignment does not correspond
with the descriptor specified 'alignment type string' alignment
The specification for PES_packet data alignment can be done at two places:
- The data_alignment_indicator from the PES_packet header.
- The alignment_type from the Data stream alignment descriptor (MPEG-2 Systems 2.6.10).

These two should agree, i.e. when the descriptor is present, the data_alignment_indicator in the PES_header must be set and another type of data alignment is found (e.g. An audio syncword is found immediately after the PES_header) then specified by the descriptor.

>>> [MPEG-2] ERROR 2436 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet data_alignment_indicator set : missing descriptor demands 'alignment type string' alignment
When the data_alignment_indicator is set in the PES_header, but no Data stream descriptor is present, an alignment_type of '01' is assumed, which is:
- Slice or video access unit alignment in case of a Video Packet
- Syncword alignment in case of an Audio Packet

This error is reported when some other form of data alignment is found.

>>> [MPEG-2] ERROR 2438 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet copyright flag set : no associated descriptor
This message is reported when the copyright flag is set to '1' in the PES_header, but a copyright descriptor, as described in MPEG-2 2.6.8, is not associated with the elementary stream which contains this PES packet.

>>> [MPEG-2] ODDITY 2439 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet copyright flag not set : but a copyright descriptor exists
This message is reported when the copyright flag is set to '0' in the PES_header, but a copyright descriptor, as described in MPEG-2 2.6.8, is associated with the elementary stream which contains this PES packet. Since the value '0' for the copyright flag does not define whether the material is copyright protected or not, this message is reported as an oddity.

>>> [MPEG-2] ERROR 2440 (ref. MPEG-2 Systems 2.4.3.7) :

PES_packet forbidden PTS_DTS_flags setting '01'

>>> [MPEG-2] ERROR 2441 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet PES_header_data_length is 'value', should be at least 'value'
The minimum length of a PES_packet header depends on the flags that are set to '1' in the PES_packet header.

>>> [MPEG-2] ERROR 2442 (ref. MPEG-2 Systems 2.7.5) :
'packet type string' PES_Packet 'number' contains no PTS although first AU starts in it
The PES_packet header did not have a PTS encoded, but the PES_packet payload contains the start of an access unit. The start of an access unit must always be accompanied with a PTS in the PES_packet header.

>>> [MPEG-2] ERROR 2445 (ref. MPEG-2 Systems 2.4.3.7) :
DTS present in low-delay video sequence's PES_packet
For Presentation units in a low_delay Video stream, the PTS should be equal to the DTS, therefore the DTS should not be encoded in the PES_packet header, because according to MPEG-2 Systems 2.7.5, a DTS may only be encoded if the decoding time differs from the presentation time.

>>> [MPEG-2] ERROR 2446 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet : DTS present for B-picture
For B-Pictures in a Video stream, the PTS should be equal to the DTS, therefore the DTS should not be encoded in the PES_packet header, because according to MPEG-2 Systems 2.7.5, a DTS may only be encoded if the decoding time differs from the presentation time.

>>> [MPEG-2] ERROR 2447 (ref. MPEG-2 Systems 2.7.5) :
PES_packet 'number' : PTS missing for the first AU after an STD-buffer underflow
in a low_delay video sequence.
When low_delay is '1', a PTS shall be encoded for the first access unit after an STD buffer underflow

>>> [MPEG-2] INFORMATION 2448 (ref. MPEG-2 Systems 2.5.2.3) :
PES low_delay STD buffer underflow at decoding time 'time string' of AU 'number'
This error reports that, according to the low_delay STD buffer calculations, one of these errors occurred:
- The AU due for removal is larger than the remaining STD buffer contents
- The size of the AU due for removal is not yet known (because the end of the AU had not been received in the stream)

>>> [MPEG-2] ERROR 2450 (ref. MPEG-2 Systems 2.7.3) :
PES_packet difference between successive ESCR's is 'time string', should be < 0.7 seconds
An ESCR should be decoded in the PES_packet at intervals of at most 0.7 seconds.

>>> [MPEG-2] ERROR 2451 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet has forbidden ES_rate value 0

>>> [MPEG-2] ERROR 2452 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet has reserved trick_mode_control 'hexadecimal value'
The trick_mode_control field uses a reserved value.

>>> [MPEG-2] ERROR 2453 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet has reserved field_id 'hexadecimal value' in its trick_mode data
The field_id field from the trick_mode uses a reserved value.

>>> [MPEG-2] ERROR 2454 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet intra_slice_refresh=0 and macroblocks are missing
The intra_slice_refresh value '0' specifies that there are no missing macroblocks between coded slices of video data in this PES_packet, so the decoder does not have to save co-sited macroblocks of previously decoded pictures. This error reports that there were missing macroblocks and this could lead to display oddities.

>>> [MPEG-2] ERROR 2455 (ref. MPEG-2 Systems 2.4.3.7) :

PES_packet has forbidden rep_cntrl value 0

>>> [MPEG-2] ERROR 2456 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet : previous_PES_packet_CRC check failed
The CRC is calculated over the data bytes of the previous PES_packet, the PES_packet header is excluded from the CRC calculation, because it can be modified during transport.

>>> [MPEG-2] ERROR 2457 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet has pack_header_field_flag set in a Program Stream
The pack_header_field_flag shall only be set to '1' in a Transport Stream, to indicate that the Pack header is encoded in the stream.

>>> [MPEG-2] ERROR 2458 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet private_data emulates a start_code_prefix 0x000001
The private_data shall be coded in such a way that this data, combined with the fields before and after, do not emulate the packet start code prefix 0x000001.

>>> [MPEG-2] ERROR 2459 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet has MPEG-1_MPEG-2_identifier=0 in an MPEG-2 Program Stream
If this flag is used in an MPEG-2 PS, it can only indicate that this (MPEG-2) packet carries data of an originally MPEG-1 PS.

>>> [MPEG-2] ERROR 2460 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet original_stuff_length 'value' larger than the allowed 'number'.

>>> [MPEG-2] ERROR 2461 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet pack_field_length 'value' should be minimal 14
The minimum length of a valid pack_header is 14 bytes.

>>> [MPEG-2] ERROR 2462 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet program_packet_sequence_counter has illegal increment ('value' -> 'value')
The program_packet_sequence_counter is an optional counter that increments with each successive PES_packet from a Program stream or from an ISO/IEC 11172-1 Stream or the PES_packets associated with a single program definition is a Transport stream. This error reports any gaps in the program_packet_sequence_counter values.

>>> [MPEG-2] ERROR 2463 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet program_packet_sequence_counter value 'value' is repeated
The program_packet_sequence_counter value is a 7 bit field, that counts successive PES_packets in a stream. The counter will wrap around to the value '0' after its maximum value. Repetition of PES_packets shall not occur, therefore, no two consecutive PES_packets in the program multiplex shall have identical program_packet_sequence_counter values.

>>> [MPEG-2] SYNTAX ERROR 2464 (ref. MPEG-2 Systems 2.4.3.6) :
PES_packet 2-bit P-STD marker is not '01'

>>> [MPEG-2] SYNTAX ERROR 2465 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet PES_extension_field_length 'value' too large
PES_header_data_length allows for only 'number'
The PES_extension_field_length increased with the already parsed bytes from the PES_header is larger than the PES_header_data_length field from the PES_packet header. This could be caused by:
- An invalid PES_extension_field_length field
- An incorrect PES_packet header flag, causing the parser to parse incorrect data.

>>> [MPEG-2] SYNTAX ERROR 2466 (ref. MPEG-2 Systems 2.4.3.7) :
PES_packet extension_field reserved byte['index number'] is 'hexadecimal value', should be 0xFF
All reserved bytes should be coded as '0xFF'.

9.2.2.3 MPEG-2 Sequence header checks

>>> [MPEG-2] ERROR 2500 (ref. MPEG-2 Video various) :
Marker_bit is 0
All market_bits should be coded as '1'.

>>> [MPEG-2] ERROR 2501 (ref. MPEG-2 Video 6.3.1) :
Sequence_header is not followed by a sequence_extension
In an MPEG-2 sequence the first Sequence_header should be followed by a Sequence_extension header.

>>> [MPEG-2] ERROR 2502 (ref. MPEG-2 Video 6.3.1) :
More than one sequence_extension specified after current sequence_header
In MPEG-2 streams, the Sequence_header should always be followed by only one Sequence_extension header.

>>> [MPEG-2] ERROR 2503 (ref. MPEG-2 Video Table 8-5) :
Sequence_scalable_extension not consistent with profile 'profile type string'
The Sequence_scalable_extension header should not be encoded for the 'Simple' and 'Main' profiles, only for the 'SNR', 'Spatial' and 'High' profiles.

>>> [MPEG-2] ERROR 2505 (ref. MPEG-2 Video 6.1.1.6) :
Unexpected sequence_display_extension
If the first Sequence_header in the sequence is not followed by an Sequence_display_extension header, all subsequent Sequence_headers should also not be followed by a Sequence_display_extension header, thus the Sequence_display_extension should not be encoded in the Sequence.

>>> [MPEG-2] ERROR 2506 (ref. MPEG-2 Video 6.3.1) :
More than one sequence_display_extension specified after current sequence_header

>>> [MPEG-2] ERROR 2508 (ref. MPEG-2 Video 6.1.1.6) :
Unexpected sequence_scalable_extension
If the first Sequence_header in the sequence is not followed by an Sequence_scalable_extension header, all subsequent Sequence_headers should also not be followed by a Sequence_scalable_extension header, thus the Sequence_scalable_extension should not be encoded in the Sequence.

>>> [MPEG-2] ERROR 2509 (ref. MPEG-2 Video 6.3.1) :
More than one sequence_scalable_extension specified after current sequence_header

>>> [MPEG-2] ERROR 2510 (ref. MPEG-2 Video 6.2.2.2.1) :
Sequence_header followed by an illegal extension
The Sequence_header is only allowed to be followed by these extensions:
- Sequence_display_extension
- Sequence_scalable_extension

>>> [MPEG-2] ERROR 2522 (ref. MPEG-2 Video Table 8-11) :
Sequence_header horizontal_size is 'value', should be <= 'maximum value' for profile/level 'profile type string'/'level type string'
The maximum horizontal size for the encoded profiles and levels must comply with the following table::

Profile	Level	Maximum horizontal size
Simple	Main / Main 422	720
Main	Main / Main 422	720
SNR scale	Main / Main 422	720
422	Main / Main 422	720
<Any>	Low	352

>>> [MPEG-2] ERROR 2527 (ref. MPEG-2 Video Table 8-11) :
Sequence_header vertical_size is 'value', should be <= 'maximum value' for profile/level level 'profile type string'/'level type string'

The maximum vertical size for the encoded profiles and levels must comply with the following table:

Profile	Level	Maximum vertical size
Simple	Main	576
Main	Main	576
SNR scale	Main	576
422	Main 422	608
<Any>	Low	288

>>> [MPEG-2] ERROR 2530 (ref. MPEG-2 Video 6.3.3) :
sequence_extension : frame_rate_code table entry exists :
frame_rate_extension values ('counter','denominator') should be 0

>>> [MPEG-2] ERROR 2531 (ref. MPEG-2 Video 6.3.3) :
sequence_extension frame_rate_extension values ('counter','denominator') have common divisor > 1

>>> [MPEG-2] ERROR 2534 (ref. MPEG-2 Video Table 8-7) :
Sequence_header : frame_rate 'value' must be 'value' when vertical_size > 'value' for profile/level 'profile type string'/'level type string'
The frame_rate for the encoded profiles and levels must comply with the following table:

Profile	Level	Vertical_size	Maximum frame_rate_code
Simple	Main	480	3 (25 Hz)
Main	Main	480	3 (25 Hz)
422	Main 422	512	3 (25 Hz)

>>> [MPEG-2] ERROR 2535 (ref. MPEG-2 Video Table 8-11) :
Sequence_header : frame_rate is 'value', must be <= 'maximum value' for profile/level 'profile type string'/'level type string'
The maximum frame_rate for the encoded profiles and levels must comply with the following table:

Profile	Level	Maximum frame_rate_code
<Any>	Main / Main 422 / Low	5 (30 Hz)

>>> [MPEG-2] ERROR 2536 (ref. MPEG-2 Video Table 8-12) :
Sequence_header : sample_rate is 'value', must be <= 'maximum value' for profile/level 'profile type string'/'level type string'
The maximum sample_rate for the encoded profiles and levels must comply with the following table:

Profile	Level	Maximum sample_rate (bit/sec)
Simple	Main	10368000
Main	Main	10368000
SNR scale	Main	10368000
Main 422	Main 422	11059200
<Any>	Low	3041280

>>> [MPEG-2] ERROR 2538 (ref. MPEG-2 Video 6.3.3) :
sequence_extension bitrate is 'value' Mbit/s, must be <= 'maximum value' Mbit/s for profile/level 'profile type string'/'level type string'
The maximum sequence_extension bitrate for the encoded profiles and levels must comply with the following table:

Profile	Level	Maximum vertical size (bit/sec)
Simple	Main	15.000.000
Main	Main	15.000.000
Main 422	Main 422	50.000.000

>>> [MPEG-2] ERROR 2539 (ref. MPEG-2 Video Table 8-14) :
sequence_extension vbv_buffer_size is 'value' KBytes, must be <= 'maximum value' KBytes for profile/level 'profile type string'/'level type string'
The maximum vbv_buffer_size for the encoded profiles and levels must comply with the following table:

Profile	Level	Maximum vbv_buffer_size (KB)
Simple	Main	224 KB = 1835008 bit
Main	Main	224 KB = 1835008 bit
Main 422	Main 422	1152 KB = 9437184 bit

>>> [MPEG-2] ERROR 2540 (ref. MPEG-2 Video 6.3.3) :
Sequence_header constrained_parameters_flag must be 0
This flag, which no longer has any meaning in MPEG-2, should never be set .

>>> [MPEG-2] WARNING 2541 (ref. MPEG-2 Video 8.1) :
This MPEG-2 (PS or TS) system stream contains a non-constrained parameters MPEG-1 video stream, which might not be decodable by some MPEG-2 decoders !

>>> [MPEG-2] ERROR 2550 (ref. MPEG-2 Video Table 8-5) :
Sequence_extension chroma_format 'value' inconsistent with specified profile
The 422 and 444 chroma_format types are only allowed in a Program Stream with a 'Main 422' or 'High' profile.

>>> [MPEG-2] ERROR 2555 (ref. MPEG-2 Video 6.3.6) :
Sequence_display_extension video_format 'value' reserved
The sequence_display_extension specified a reserved video_format value, i.e. a value larger than 0x6.

>>> [MPEG-2] ERROR 2556 (ref. MPEG-2 Video 6.3.6) :
Sequence_display_extension colour_primaries 'value' reserved
The sequence_display_extension specified a reserved colour_primaries value, i.e. a value larger than 0x8.

>>> [MPEG-2] ERROR 2557 (ref. MPEG-2 Video 6.3.6) :
Sequence_display_extension transfer_charact. 'value' reserved
The sequence_display_extension specified a reserved transfer_characteristics value, i.e. a value larger than 0x9.

>>> [MPEG-2] ERROR 2580 (ref. MPEG Video 8 (before 8.1)) :
Sequence_extension profile_and_level_indication is 'hexadecimal value'
Illegal profile and/or level is specified.

>>> [MPEG-2] ERROR 2581 (ref. MPEG Video 6.3.5) :
Sequence_extension profile_and_level_indication is 'value', previously defined to be 'previous value'
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2582 (ref. MPEG Video 6.3.5) :
Sequence_extension progressive_sequence is 'value', previously defined to be 'previous value'
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2583 (ref. MPEG Video 6.3.5) :
Sequence_extension chroma_format is 'value'
The sequence_display_extension specified a reserved chroma_format value, i.e. the value 0x0.

>>> [MPEG-2] ERROR 2584 (ref. MPEG Video 6.3.5) :

Sequence_extension chroma_format is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2585 (ref. MPEG Video 6.3.3 & 6.3.5) :
Sequence_extension horizontal_size is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] SYNTAX ERROR 2586 (ref. MPEG Video 6.3.5) :
Sequence_extension marker bit is 0

>>> [MPEG-2] ERROR 2587 (ref. MPEG Video 6.3.3 & 6.3.5) :
Sequence_extension vertical_size is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2588 (ref. MPEG Video 6.3.3 & 6.3.5) :
Sequence_extension bit_rate is 0

>>> [MPEG-2] ERROR 2589 (ref. MPEG Video 6.3.3 & 6.3.5) :
Sequence_extension bit_rate is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2590 (ref. MPEG Video 6.3.3 & 6.3.5) :
Sequence_extension buffer size is ‘value’ KBytes, previously defined to be ‘previous value’ KBytes
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2591 (ref. MPEG Video 6.3.5) :
Sequence_extension low_delay is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2592 (ref. MPEG Video 6.3.5) :
Sequence_extension frame_rate_extension_n is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2593 (ref. MPEG Video 6.3.5) :
Sequence_extension frame_rate_extension_d is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2600 (ref. MPEG Video 6.3.6) :
Sequence_display_extension video_format is ‘value’, previously defined to be ‘value’
All of the fields in repeated Sequence_display_extension headers shall have the same values as the first Sequence_display_extension header in the stream.

>>> [MPEG-2] ERROR 2601 (ref. MPEG Video 6.3.6) :
Sequence_display_extension colour_description is ‘value’, previously defined to be ‘previous value’
All of the fields in repeated Sequence_display_extension headers shall have the same values as the first Sequence_display_extension header in the stream.

>>> [MPEG-2] ERROR 2602 (ref. MPEG Video 6.3.6) :
Sequence_display_extension colour_primaries is ‘value’, previously defined to be ‘previous value’

All of the fields in repeated sequence_extension headers shall have the same values as the first sequence_extension header in the stream.

>>> [MPEG-2] ERROR 2603 (ref. MPEG Video 6.3.6) :
Sequence_display_extension transfer_characteristics is 'value', previously defined to be 'previous value'
All of the fields in repeated Sequence_display_extension headers shall have the same values as the first Sequence_display_extension header in the stream.

>>> [MPEG-2] ERROR 2604 (ref. MPEG Video 6.3.6) :
Sequence_display_extension matrix_coefficients is 'value'
The sequence_display_extension specified a reserved matrix_coefficients value, i.e. the value 0x0 or a value larger than 0x8.

>>> [MPEG-2] ERROR 2605 (ref. MPEG Video 6.3.6) :
Sequence_display_extension matrix_coefficients is 'value', previously defined to be 'previous value'
All of the fields in repeated Sequence_display_extension headers shall have the same values as the first Sequence_display_extension header in the stream.

>>> [MPEG-2] ERROR 2606 (ref. MPEG Video 6.3.6) :
Sequence_display_extension display_horizontal_size is 'value', previously defined to be 'previous value'
All of the fields in repeated Sequence_display_extension headers shall have the same values as the first Sequence_display_extension header in the stream.

>>> [MPEG-2] ERROR 2607 (ref. MPEG Video 6.3.6) :
Sequence_display_extension display_vertical_size is 'value', previously defined to be 'previous value'
All of the fields in repeated Sequence_display_extension headers shall have the same values as the first Sequence_display_extension header in the stream.

9.2.2.4 MPEG-2 GOP checks

>>> [MPEG-2] ERROR 2621 (ref. MPEG-2 Video 6.3.1) :
GOP header is followed by extension_data

9.2.2.5 MPEG-2 Picture checks

>>> [MPEG-2] ERROR 2650 (ref. MPEG-2 Video 6.3.1) :
Picture is not followed by a picture_coding_extension
A picture in an MPEG-2 compliant stream must always be followed by a picture_extension header. The parser will recover until the next start code.

>>> [MPEG-2] ERROR 2651 (ref. MPEG-2 Video 6.3.1) :
More than one picture_coding_extension specified after current picture_header

>>> [MPEG-2] ERROR 2652 (ref. MPEG-2 Video 6.3.1) :
More than one quant_matix_extension specified after current picture_header

>>> [MPEG-2] ERROR 2653 (ref. MPEG-2 Video 6.3.12) :
Picture_display_extension only allowed when a sequence_display_extension present

>>> [MPEG-2] ERROR 2654 (ref. MPEG-2 Video 6.3.1) :
More than one Picture_display_extension specified after current picture_header

>>> [MPEG-2] ERROR 2655 (ref. MPEG-2 Video Table 8-5) :
Picture_scalable_extension not consistent with profile 'profile type string'
A Picture_scalable_extension is not allowed to be encoded in Program Streams with the following profiles:
- Simple profile

- Main profile
- SNR scalable profile

>>> [MPEG-2] ERROR 2656 (ref. MPEG-2 Video 6.3.1) :
More than one picture_spatial_scalable_extension specified after current picture_header

>>> [MPEG-2] ERROR 2657 (ref. MPEG-2 Video 6.3.1) :
More than one picture_temporal_scalable_extension specified after current picture_header

>>> [MPEG-2] ERROR 2658 (ref. MPEG-2 Video 6.3.1) :
More than one copyright_extension specified after current picture_header

>>> [MPEG-2] ERROR 2659 (ref. MPEG-2 Video 6.2.2.2.1) :
Picture_header followed by an illegal extension
A Picture header is allowed to be followed by one of these extensions:
- Copyright_extension
- Picture temporal scalable extension
- Picture spatial scalable extension
- Picture display extension
- Quantiser matrix extension
- Picture coding extension

The parser did not find any of these extensions after the Picture header and will recover to the next valid start code.

>>> [MPEG-2] ERROR 2673 (ref. MPEG-2 Video Table 8-5) :
B-pictures are not permitted with Simple Profile

>>> [MPEG-2] ERROR 2675 (ref. MPEG-2 Video 6.1.1.11) :
First picture after a sequence header must be either an I- or a P-picture

>>> [MPEG-2] ERROR 2676 (ref. MPEG-2 Video 6.3.9) :
Picture_header picture_coding_type of both frame pictures should be the same
In case of pictures that are encoded using 2 interlaced frame pictures, the picture_coding_type in the Picture_header must describe the same value for each frame of the picture.

>>> [MPEG-2] ERROR 2677 (ref. MPEG-2 Video 6.3.10) :
Picture_header picture_coding_type of the 2nd frame picture should be I or P
When a frame is encoded as two field pictures, both fields must be of the same picture_coding_type, except when the 1st encoded field is an I-picture (then the 2nd may be either an I- or P-picture).

>>> [MPEG-2] ERROR 2678 (ref. MPEG-2 Video 6.3.5) :
Picture_header : low_delay sequence does not allow B-pictures

>>> [MPEG-2] ERROR 2681 (ref. MPEG-2 Video 6.3.9) :
Picture_header full_pel_'forward|backward'_vector should be 0
The full_pel_backward_vector must be '0' for a B-picture, the full_pel_forward_vector must be '0' for both B-pictures and P-pictures.

>>> [MPEG-2] ERROR 2682 (ref. MPEG-2 Video 6.3.9) :
Picture_header 'forward|backward'_f_code should be 0x7
The backward_f_code must be '0x7' for a B-picture, the forward_f_code must be '0x7' for both B-pictures and P-pictures.

>>> [MPEG-2] ERROR 2690 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension 'forward|backward'_'horizontal|vertical'_f_code has the forbidden value 0

>>> [MPEG-2] ERROR 2691 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension 'forward|backward'_'horizontal|vertical'_f_code has the reserved value 'value'
One of these fields specified a reserved _f_code, i.e. a value in the range [10..14]:
- forward_horizontal_f_code
- forward_vertical_f_code
- backward_horizontal_f_code
- backward_vertical_f_code

>>> [MPEG-2] ERROR 2692 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension 'forward|backward'_'horizontal|vertical'_f_code should be 0xF
The value '0xF' should be encoded for the specified _f_code in these cases:
- The picture type is 'I' and the concealment_motion_vectors field from the Picture_coding_extension equals '0'.
- The picture type is either 'I' or 'P' and the specified _f_code is a forward_.._f_code.

>>> [MPEG-2] ERROR 2694 (ref. MPEG-2 Video Table 8-8) :
Picture_coding_extension 'forward|backward'_'horizontal|vertical'_f_code is 'value', must be <= 'maximum value' for 'level type string' level
This error is reported when:
- The stream uses the Low level encoding scheme and the .._horizontal_f_code exceeds '7'.
- The stream uses the Main level encoding scheme and the .._horizontal_f_code exceeds '8'.
- The stream uses the Low level encoding scheme with frame pictures _f_code exceeds '4'.
- The stream uses the Main level encoding scheme with frame pictures _f_code exceeds '5'.
- The stream uses the Low level encoding scheme without frame pictures _f_code exceeds '3'.
- The stream uses the Main level encoding scheme without frame pictures _f_code exceeds '4'.

>>> [MPEG-2] ERROR 2696 (ref. MPEG-2 Video Table 8-5) :
Picture_coding_extension intra_dc_precision 'value' inconsistent with profile 'profile type string'
This error is reported when the intra_dc_precision value is not encoded as '11b' with these profiles:
- Simple profile
- Main profile
- SNR scalable profile
- Spatial scalable profile

>>> [MPEG-2] ERROR 2700 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension picture_structure value '0' reserved

>>> [MPEG-2] ERROR 2701 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension picture_structure of a frames's 2nd field ('value') must be of opposite parity
If the picture_structure from the Picture_coding_extension describes the field type of the picture, this field type must alternate between the TOP field and BOTTOM field values.

>>> [MPEG-2] ERROR 2705 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension top_field_first for a field_picture should be 0

>>> [MPEG-2] ERROR 2706 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension top_field_first for repeat_first_field=1 should be 0.

>>> [MPEG-2] ERROR 2707 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension frame_pred_frame_dct for a field_picture should be 0

>>> [MPEG-2] ERROR 2708 (ref. MPEG-2 Video Corrigendum: item 5) :
Picture_coding_extension frame_pred_frame_dct should be 1 when progressive_sequence is 1

>>> [MPEG-2] ERROR 2709 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension repeat_first_field for a field_picture should be 0

>>> [MPEG-2] ERROR 2710 (ref. MPEG-2 Video 6.3.10) :

Picture_coding_extension repeat_first_field for non-progressive_frame should be 0

>>> [MPEG-2] ERROR 2711 (ref. MPEG-2 Video Table 8-7) :
Picture_coding_extension repeat_first_field for PAL B-picture should be 0

>>> [MPEG-2] ERROR 2712 (ref. MPEG-2 Video Table 8-7) :
Picture_coding_extension repeat_first_field does not satisfy the constraints

>>> [MPEG-2] ERROR 2713 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension chroma_420_type should be 0

>>> [MPEG-2] ERROR 2714 (ref. MPEG-2 Video 6.3.10) :
Picture_coding_extension chroma_420_type must be the same as progressive_frame

>>> [MPEG-2] ERROR 2720 (ref. MPEG Video 6.3.11) :
Quant_matrix_extension ‘type’intra_quantiser_matrix[‘index’] is 0

>>> [MPEG-2] ERROR 2721 (ref. MPEG Video 6.3.11) :
Quant_matrix_extension ‘type’intra_quantiser_matrix[0] is ‘value’, should be 8

>>> [MPEG-2] ERROR 2723 (ref. MPEG Video 6.3.11) :
Quant_matrix_extension load_chroma_’type’intra_quantiser_matrix is 1, should be 0 when chroma-format is 4:2:0

>>> [MPEG-2] ERROR 2724 (ref. MPEG Video 6.3.15) :
Copyright_extension copyright_identifier should be 0 when copyright_flag is 0

>>> [MPEG-2] ERROR 2725 (ref. MPEG Video 6.3.15) :
Copyright_extension reserved should be 0

>>> [MPEG-2] WARNING 2726 (ref. MPEG Video 2-C.1 | Ann.C) :
VBV buffer underflow for picture ‘number’ (‘number’ bytes, m=’value’)

>>> [MPEG-2] ERROR 2727 (ref. MPEG Video 2-C.1 | Ann.C) :
Splicing point VBV buffer overflow for picture ‘number’ (‘number’ bytes, m=’value’)

9.2.2.6 MPEG-2 Slice checks

>>> [MPEG-2] ERROR 2751 (ref. MPEG Video 6.3.16) :
Slice vertical position ‘value’, vertical position of previous slice is ‘value’, the difference between the vertical position of current slice and the previous one should be 0 or 1

This error is reported when the slice_vertical_position is smaller or more than 1 larger then the previous slice_vertical_position. Since Slices shall occur in the bit stream in the order in which they are encountered, the slice_vertical_position cannot be smaller than the previous slice_vertical_position and since the first and last macroblock of a slice shall be in the same horizontal row of macroblocks, the Slice is always equal to or less than 1 complete line.

>>> [MPEG-2] ERROR 2752 (ref. MPEG-2 Video 6.3.16) :
Slice vertical_position (‘value’) should be < 128

The slice_vertical_position shall be in the range [1:128] when the slice_vertical_position_extension is present in the Program Stream.

>>> [MPEG-2] WARNING 2759 (ref. MPEG-2 Video 6.3.16) :
Slice intra_slice set, but non-intra macroblocks occur

The intra_slice flag shall be set to '0' if any of the macroblocks in the slice are non-intra macroblocks. This flag may only be set to '1' when all of the macroblocks in the Slice are intra macroblocks.

>>> [MPEG-2] ERROR 2761 (ref. MPEG-2 Video Corrigendum: item 9) :
Slice slice_picture_id should be 0 when slice_picture_id_enable is 0

>>> [MPEG-2] ERROR 2762 (ref. MPEG-2 Video Corrigendum: item 9) :
Slice slice_picture_id_enable is ‘value’, previously defined to be ‘previous value’
The slice_picture_id_enable shall have the same value as encoded for the slice_picture_id_enable from the first slice of a picture.

>>> [MPEG-2] ERROR 2763 (ref. MPEG-2 Video Corrigendum: item 9) :
Slice slice_picture_id is ‘value’, previously defined to be ‘previous value’
The slice_picture_id shall have the same value as encoded for the slice_picture_id from the first slice of a picture.

9.2.2.7 MPEG-2 Macroblock checks

>>> [MPEG-2] ERROR 2771 (ref. MPEG-2 Video 6.3.17) :
Macroblock stuffing illegal
Macroblock stuffing is not allowed in MPEG-2 video streams.

>>> [MPEG-2] ERROR 2772 (ref. MPEG-2 Video 6.3.17.1) :
Macroblock frame_motion_type value ‘value’ reserved
The frame_motion_type in the Macroblock specified a reserved value, i.e. '0x0'

>>> [MPEG-2] ERROR 2773 (ref. MPEG-2 Video 6.3.17.1) :
Macroblock field_motion_type value ‘value’ reserved
The field_motion_type in the Macroblock specified a reserved value, i.e. '0x0'

>>> [MPEG-2] ERROR 2774 (ref. MPEG-2 Video Table B-9) :
Coded_block_pattern is 0 when chroma_format is 4:2:0
The coded_block_pattern value '0' shall not be used with 4:2:0 chrominance structure.

>>> [MPEG-2] ERROR 2775 (ref. MPEG-2 Video 7.6.3.5) :
Macroblock non-frame prediction within a P frame picture in case macroblock_motion_forward is zero and macroblock_intra is zero as well
The prediction_type in the macroblocks belonging to a P frame picture with macroblock_motion_forward and macroblock_intra equalling '0', should be "frame-based", i.e. '10b'.

>>> [MPEG-2] ERROR 2776 (ref. MPEG-2 Video 7.6.1) :
Macroblock non-field prediction within a P field picture in case macroblock_motion_forward is zero and macroblock_intra is zero as well
The prediction_type in the macroblocks belonging to a P field picture with macroblock_motion_forward and macroblock_intra equalling '0', should be "field-based", i.e. '01b'.

>>> [MPEG-2] ERROR 2778 (ref. MPEG-2 Video 7.6.1) :
Macroblock dual-prime prediction with a B-picture between predicted & reference pictures
The Dual-prime prediction mode may only be used in P-pictures (field or frame encoded) when there are no B-pictures between the predicted and reference fields or frames.

>>> [MPEG-2] ERROR 2780 (ref. MPEG-2 Video 7.6.3.5) :
Macroblock dual-prime prediction illegal in a 2nd P-field of an I-frame
In the case that a P field picture is used as the second field of a frame, in which the first field is an I field picture, the Dual-prime prediction mode shall not be used. This ensures that prediction is only made from the I field picture.

>>> [MPEG-2] ERROR 2785 (ref. MPEG-2 Video 7.6.3.5) :

Macroblock macroblock_motion_forward=0 & macroblock_intra=0 combination is illegal in a 2nd P-field of an I-frame

In the case that a P field picture is used as the second field of a frame, in which the first field is an I field picture, the macroblock_motion_forward and macroblock_intra shall be encoded as '0'.

>>> [MPEG-2] ERROR 2786 (ref. MPEG-2 Video 7.6.3.5) :

Skipped macroblocks not allowed in a 2nd P-field of an I-frame

In the case that a P field picture is used as the second field of a frame, in which the first field is an I field picture, there shall be no skipped macroblocks.

>>> [MPEG-2] ERROR 2787 (ref. MPEG-2 Video 7.6.3.5) :

Macroblock motion_vertical_field_select has the same parity as the field being predicted, in a 2nd P-field of an I-frame

In the case that a P field picture is used as the second field of a frame, in which the first field is an I field picture, the motion_vertical_field_select shall not indicate the same parity as the field being predicted. This ensures that prediction is only made from the I field picture.

>>> [MPEG-2] ODDITY 2796 (ref. MPEG-2 Video 7.6.3.9) :

Vertical component of concealment vector of in last row of picture greater than 0

For all macroblocks, concealment motion vectors should be appropriate for use in the macroblock that lies vertically below the macroblock in which the motion vector occurs. Since the bottom row of macroblocks have no macroblocks that lie vertically below them, the vertical component of concealment vector should be encoded as '0'. This message is reported as an ODDITY.

>>> [MPEG-2] ERROR 2797 (ref. MPEG-2 Video 8.2) :

More than 2 macroblocks exceed the max. number of bits ‘number’

Only 2 macroblocks in each horizontal row of macroblocks may exceed the following size:

chroma_format	maximum number of bits
4:2:0	4608
4:2:2	6144
4:4:4	9216

9.2.2.8 MPEG-2 Audio checks

>>> [MPEG-2] ERROR 2851 :
Base frame data ‘number’ bytes ahead of extension frame data (allowed 4096 bytes)
The base and extension streams for Audio must be multiplexed in such a way that the base frame and the associated extension frame are no more than 4096 bytes apart.

>>> [MPEG-2] ERROR 2852 :
Extension frame data ‘number’ bytes ahead of base frame data (allowed 4096 bytes)
The base and extension streams for Audio must be multiplexed in such a way that the base frame and the associated extension frame are no more than 4096 bytes apart.

>>> [MPEG-2] SYNTAX ERROR 2855 (ref. MPEG-2 Audio) :
Specified number of ancillary data bytes (‘value’) does not fit in frame
The specified n_ad_bytes, together with the already parsed bytes from the Audio base frame, should be maximum 1152 bytes long. This error could indicate a n error in the n_ad_bytes field or with the encoding of the other fields in the base frame.

>>> [MPEG-2] ERROR 2856 (ref. MPEG-2 Audio 2.5.3.1) :
Not all fields of mc_header fit in base frame
In case of an MPEG-2 multichannel Audio stream, the base frame should consist of the complete MPEG-1 audio data and the complete MPEG-2 multichannel header. This error reports that some fields of the multichannel header could not be parsed before the end of the base frame. This is usually caused by decoding an MPEG-1 Audio stream with the standard MPEG-2 setting of the parser.

>>> [MPEG-2] ERROR 2857 (ref. MPEG-2 Audio 2.5.2.13) :
Centre value '10b' is not defined.
The value ‘10b’ or the centre field is not defined and therefore not to be used.

>>> [MPEG-2] ERROR 2858 (ref. MPEG-2 Audio 0.2.3.2) :
In ‘mode type string’ mode, no centre channel allowed
These modes cannot specify a centre channel:
- Single channel
- Dual channel

>>> [MPEG-2] ERROR 2859 (ref. MPEG-2 Audio 0.2.3.2) :
In ‘mode type string’ mode, no ‘surround type string’ allowed
These modes cannot specify a mono or stereo surround channel:
- Single channel
- Dual channel

>>> [MPEG-2] ERROR 2860 (ref. MPEG-2 Audio 0.2.3.2) :
In ‘mode type string’ mode, no lfe allowed
An Audio stream in 1/0 configuration (single channel, without a second 2/0 stereo programme) cannot specify a low frequency enhancement channel.

>>> [MPEG-2] ERROR 2861 (ref. MPEG-2 Audio 2.5.2.13) :
Dematrix_procedure value '10' is only allowed in 3/1 or 3/2 configuration

>>> [MPEG-2] ERROR 2862 (ref. MPEG-2 Audio 2.5.2.15) :
Tc_allocation value ‘value’ exceeds maximum allowed value ‘maximum value’ for current configuration
The Tc_allocation is restricted according to:

Surround mode \ Centre channel	yes	no
None	2	-
Mono	5	2
Stereo	7	3
Second programme	2	-

>>> [MPEG-2] ERROR 2864 (ref. MPEG-2 Audio 2.5.2.15) :
Tc_allocation value ‘value’ is not allowed if Phantom coding is used (centre == '11')
In case of Phantom coding of the centre (centre = ‘11b’) channel, the Tc_allocation is should not specify the values ‘5’ (when the surround mode equals Stereo Surround (surround = ‘10b’)) and ‘1’ and ‘2’.

>>> [MPEG-2] ERROR 2865 (ref. MPEG-2 Audio 2.5.2.15) :
Dyn_cross_mode has forbidden value ‘value’ for current configuration
The Dyn_cross_mode is restricted according to:

Surround mode \ Centre channel	yes	no
None	1	-
Mono	4	1
Stereo	14	4
Second programme	1	-

>>> [MPEG-2] ERROR 2866 (ref. MPEG-2 Audio 2.5.2.10) :
Ext_header has reserved ext_ID_bit value ‘value’
The ext_ID_bit should be set to ‘0’, as it is reserved for future use.

9.3 DVD CHECKS

These messages relate to DVD-Video application checks.

9.3.1 DVD System checks

>>> [DVD] ERROR 3001 (ref. DVD-3 2.1) :
ERR_DVD_SRSV_0
Reserved bits shall be all 0.

>>> [DVD] ERROR 3002 (ref. N/A) :
ERR_DVD_ILLEGAL_ILVU
An illegal ILVU has been found. Since only Angle blocks are supported, this ILVU would be the 'number'-th Angle, while only 'number' Angles defined in the current Title. Possible causes:
- The Title specified the wrong number of Angles
- An unsupported Parental block in the stream
- An unsupported Language credit block in the stream
Parsing is stopped!!!!

>>> [DVD] ERROR 3005 (ref. N/A) :
ERR_DVD_NO_XCHECK_PARAS
Necessary cross check parameters not found on the cross check data file! Default values are used for missing cross check parameters.

>>> [DVD] SYNTAX ERROR 3009 :
ERR_DVD_SYNTAX_RECOVER
Parsing impossible due to syntax error : data skipped

9.3.2 DVD VOB checks

>>> [DVD] INFORMATION 3011 :
ERR_DVD_NEW_VOB
New VOB start!

>>> [DVD] INFORMATION 3012 :
ERR_DVD_NEW_CELL
New Cell start !

>>> [DVD] INFORMATION 3013 :
ERR_DVD_NEW_ILVB
New Interleaved Block start !

>>> [DVD] INFORMATION 3014 :
ERR_DVD_NEW_ILVU
New Interleaved Unit start !

>>> [DVD] ERROR 3015 (ref. DVD-3 Table 5.1-1) :
ERR_DVD_1ST_VOBU_VID
The first VOBU of a VOB should have the video data

>>> [DVD] ODDITY 3020 :
ERR_DVD_VOBU_EMPTY
The previous VOBU does not contain any data !

>>> [DVD] ERROR 3022 (ref. DVD-3 5.1.1) :

ERR_DVD_VOBU_MIN_LEN
A VOBU represents a presentation period of at least 0.4 seconds.

>>> [DVD] ERROR 3023 (ref. DVD-3 5.1.1) :
ERR_DVD_VOBU_MAX_LEN
A VOBU except the last VOBU of a cell shall represent a presentation period of at most 1 second. The last VOBU of a cell shall represent a presentation period of at most 1.2 seconds.

>>> [DVD] ERROR 3025 (ref. DVD-3 5.1.1 rule 1) :
ERR_DVD_VOBU_PERIOD
The presentation period of a VOBU is equal to an integer number of video field periods. This is also the case when the VOBU does not contain any video data.

>>> [DVD] ERROR 3027 (ref. DVD-3 5.1.1 rule 2) :
ERR_DVD_VOBU_START
The presentation start and termination time of a VOBU are defined in 90 kHz units. The presentation start time of a VOBU is equal to the presentation termination time of the previous VOBU. (except for the first VOBU).

>>> [DVD] WARNING 3031 (ref. DVD-3 5.1.1) :
ERR_DVD_VOBU_PTS_VOBU_START
The current VOBU contains a PTS, which is more than a video field period earlier than the VOBU presentation start time.

>>> [DVD] WARNING 3032 (ref. DVD-3 5.1.1) :
ERR_DVD_VOBU_PTS_VIDEO_START
The current VOBU contains a PTS, which is more than a video field period earlier than its video presentation start time.

>>> [DVD] WARNING 3033 (ref. DVD-3 5.1.1) :
ERR_DVD_VOBU_PTS_VOBU_END
The current VOBU contains a PU with a presentation time, which is more than a video field period later than the VOBU presentation termination time.

>>> [DVD] ERROR 3041 (ref. DVD-3 5.1.1 rule 5) :
ERR_DVD_VOBU_NOVID_NO_SEQ_END
When a VOBU with video data is followed by a VOBU without video data (in the same VOB), the last coded picture must be followed by a sequence_end_code.

>>> [DVD] ERROR 3042 (ref. DVD-3 5.1.1 rule 6) :
ERR_DVD_VOBU_LONG_NO_SEQ_END
When the presentation period of the VOBU is longer than the presentation period of the video it contains, the last coded picture must be followed by a sequence_end_code.

>>> [DVD] ERROR 3043 (ref. DVD-3 5.1.1 rule 7) :
ERR_DVD_VOBU_MULTI_SEQ_END
The video data in a VOBU must never contain more than one sequence_end_code.

>>> [DVD] ERROR 3046 (ref. DVD-3 5.4.1) :
ERR_DVD_VOBU_NO_SEQ_HDR
A VOBU's video data must start with a sequence_header.

>>> [DVD] ERROR 3047 (ref. DVD-3 5.4.1) :
ERR_DVD_VOBU_NO_GOP_HDR
A VOBU's video data must have a GOP_header following the sequence_header at the start.

>>> [DVD] ERROR 3048 (ref. DVD-3 5.4.1) :

ERR_DVD_VOBU_NO_I_PIC
A VOBU's video data must have an I-picture following the sequence_ & GOP_header at the start.

9.3.3 DVD Pack checks

>>> [DVD] ERROR 3101 (ref. DVD-3 5.2.1) :
ERR_DVD_PACK_LEN
The Pack length must be 2048 bytes.

>>> [DVD] ERROR 3103 (ref. DVD-3 5.2.1) :
ERR_DVD_PAD_NOTLAST
Padding packet must be last in any pack.

>>> [DVD] ERROR 3106 (ref. DVD-3 Table 5.2.1-2) :
ERR_DVD_SCR_32
SCR_base[32] must be 0.

>>> [DVD] ERROR 3107 (ref. DVD-3 3.3.12.4) :
ERR_DVD_SCR_0
SCR in the first pack of each VOB must be 0.

>>> [DVD] ERROR 3108 (ref. DVD-3 Table 5.2.1-2) :
ERR_DVD_MUXRATE
Pack program_mux_rate must be set to 10.08 Mbps

>>> [DVD] ERROR 3109 (ref. DVD-3 Table 5.2.1-2) :
ERR_DVD_STUFLEN
Pack stuffing_length must be 0. The pack length adjustment method for DVD is:

Number of adjusted data	**Adjustment method**
1 to 7 bytes	Insert stuffing bytes in packet header
8 bytes or more	Add a padding packet as the last packet in a pack

>>> [DVD] ERROR 3111 (ref. DVD-3 5.2.2) :
ERR_DVD_NV_PCK_NOT1ST
The navigation pack must be aligned to the first pack of the VOBU.

>>> [DVD] ERROR 3112 (ref. DVD-3 5.2.2) :
ERR_DVD_NV_PCK_NO_SYSPCIDSI
The Navigation pack comprises a pack header, a system header, a PCI packet and a DSI packet.

>>> [DVD] ERROR 3113 (ref. DVD-3 5.2.2) :
ERR_DVD_ILL_IN_PCK
The Navigation pack may only contain a pack header, a system header, a PCI packet and a DSI packet.

>>> [DVD] ERROR 3121 (ref. DVD-3 Table 5.2.2-1 Note 1) :
ERR_DVD_PACKET_RATE
Only the packet rate of the NV_PCK and the MPEG-2 audio format 2 pack may exceed the packet rate defined in the “Constrained system parameter Program stream” of the ISO/IEC 13818-1.

>>> [DVD] ERROR 3123 (ref. DVD-3 Table 5-2 *1) :
ERR_DVD_SP_TRANSF_RATE

	transfer rate		Note
	Total streams	One stream	
VOB	10.8 Mbps	---	

Video stream	9.80 Mbps	9.80 Mbps	Number of streams = 1
Audio streams	9.80 Mbps	6.144 Mbps	Number of streams = 8 (max)
Sub-picture streams	9.80 Mbps	3.36 Mbps *1	Number of streams = 32 (max)

*1 The restriction on Sub-picture stream in a VOB shall be define by the following rule:
a) For all Sub-picture packs which have the same sub-stream_id (SP_PCK(i)):

$SCR(n) \leq SCR(n+10) - T_{30packs}$

Where
n : 1 to (number of SP_PCK(i)s - 10)
SCR(n) : SCR of the n-th SP_PCK(i)
SCR(n+10) : SCR of the 10^{th} SP_PCK(i) after the n-th SP_PCK(i)
$T_{30packs}$: value of 1316571 ($=27 \times 10^6 \times 30 \times 2048 \times 8 / 10.08 \times 10^6$)

b) For all Sub-picture packs (SP_PCK(all) in a VOB which may be connected seamlessly with the succeeding VOB:

$SCR(n) \leq SCR(last) - T_{9packs}$

Where
n : 1 to (number of SP_PCK(all)s)
SCR(n) : SCR of the n-th SP_PCK(all)
SCR(last) : SCR of the last pack in the VOB
T_{9packs} : value of 394971 ($=27 \times 10^6 \times 30 \times 2048 \times 9 / 10.08 \times 10^6$)

Note: At least the first pack of the succeeding VOB is not SP_PCK. T_{9packs} plus $T_{1stpack}$ guarantee ten successive packs.

>>> [DVD] INFORMATION 3124 :
ERR_DVD_EMPTY_PACK
Gives information about data in the pack.

9.3.4 DVD System header checks

>>> [DVD] ERROR 3151 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_BOUND_ERR
The system_header's
- audio_bound must be between 0 and 8
- video_bound must be 1

>>> [DVD] ERROR 3152 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_NOT_FIXED
The system_header's fixed_flag must be 0 (variable bit rate).

>>> [DVD] ERROR 3153 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_CSPS_FLAG
The system_header's CSPS_flag must be 0.

>>> [DVD] ERROR 3154 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_LOCK_FLAG_0
The system_header's system_audio_lock_flag and system_video_lock_flag must be 1.

>>> [DVD] ERROR 3155 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_STRID_ILL
In the system_header:
- the stream_id for all Video must be 1011 1001b (0xB9)
- the stream_id for all Audio must be 1011 1000b (0xB8)
- the stream_id for private_1 must be 1011 1101b (0xBD)
- the stream_id for private_2 must be 1011 1111b (0xBF)

>>> [DVD] ERROR 3156 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_STRID_AV
In the system_header:
- the stream_id for all Video must be 1011 1001b (0xB9)
- the stream_id for all Audio must be 1011 1000b (0xB8)
- the stream_id for private_1 must be 1011 1101b (0xBD)
- the stream_id for private_2 must be 1011 1111b (0xBF)

>>> [DVD] ERROR 3157 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_STRID_MISS
All four entries (all Video, all Audio, private_1, private_2) must appear in the system_header.

>>> [DVD] WARNING 3159 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_STRID_ORDER
The order of the entries in the system_header must be
1. all Video streams
2. all Audio streams
3. private_stream_1
4. private_stream_2

>>> [DVD] ERROR 3161 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_BUF_SCALE
The system_header's P-STD_buf_bound_scale must be
- buf_size x 1024 bytes for all video streams
- buf_size x 128 bytes for all audio streams
- buf_size x 1024 bytes for private_stream_1
- buf_size x 1024 bytes for private_stream_2

>>> [DVD] ERROR 3162 (ref. DVD-3 Table 5.2.2-1) :
ERR_DVD_BUF_BOUND
The system_header's P-STD_buf_size_bound must be

- 237568 bytes for all video streams
- 4096 bytes for all audio streams
- 59392 bytes for private_stream_1
  The sum of the target buffers for the presentation data defined as private_stream_1 shall be described.
- 2048 bytes for private_stream_2

9.3.5 DVD Packet checks

>>> [DVD] ERROR 3201 (ref. DVD-3 Table 5.2.3-1 e.f.) :
ERR_DVD_PKT_FLAG_1
The following flags must be 0:

- ESCR_flag
- ES_rate_flag
- DSM_trick_mode_flag
- additional_copy_info_flag
- PES_CRC_flag
- PES_private_data_flag
- pack_header_field_flag
- program_packet_sequence_counter_flag
- PES_extension_flag_2

>>> [DVD] ERROR 3202 (ref. DVD-3 Table 5.2.3-1 e.f.) :
ERR_DVD_PKT_FLAG_0
P-STD_buffer_flag must be 1.

>>> [DVD] ERROR 3203 (ref. DVD-3 Table 5.2.3-1 e.f.) :
ERR_DVD_PKT_ILL_SCR_CTRL
PES_scrambling_control must be 0 (or 1 when scrambled).

>>> [DVD] ERROR 3204 (ref. DVD-3 Table 5.2.3-1 e.f.) :
ERR_DVD_PKT_ILL_PSTDTSFLS
PTS_DTS_flags value must be 00b, 10b or 11b.

>>> [DVD] ERROR 3206 (ref. DVD-3 Table 5.2.3-1 e.f. Note 2) :
ERR_DVD_PKT_PES_EXT
PES_extension is only allowed for the first packet of a VOB.

>>> [DVD] ERROR 3207 (ref. DVD-3 Table 5.2.3-1 e.f. Note 2) :
ERR_DVD_PKT_PES_MISSING
PES_extension is expected for the first packet of a VOB.

>>> [DVD] ERROR 3209 (ref. DVD-3 Table 5.2.3-1 e.f.) :
ERR_DVD_PKT_HDRDAT_LEN
PES_header_data_length value shall be between 0 and 20.

>>> [DVD] ERROR 3210 (ref. DVD-3 Table 5.2.3-1 e.f. Note 1) :
ERR_DVD_PKT_PDTS_32
PTS[32] and DTS[32] shall be set to zero.

>>> [DVD] ERROR 3211 (ref. DVD-3 Table 5.2.3-1 e.f.) :
ERR_DVD_PKT_STDBUF_SCALE
P_STD_buffer_scale value shall be set to 1.

>>> [DVD] ERROR 3212 (ref. DVD-3 Table 5.2.3-1 | 5.2.4-3) :
ERR_DVD_PKT_STDBUF_SIZE
P_STD_buffer_size value shall be set to:
- 232 (payload according to ISO 13818-2)
- 46 (payload according to ISO 11172-2)

>>> [DVD] ERROR 3213 (ref. DVD-3 Table 5.2.4-1,2 Note 1 | 5.2.5-1 Note 1) :
ERR_DVD_PKT_STDBUF_MAX_SIZE
PES_packet has a too large P_STD_buffer_size value 'number', should be smaller than 'number' bytes.

>>> [DVD] INFORMATION 3214 (ref. MPEG2 2.4.3.6/7 | DVD-3 Table 5.2.3-1) :
ERR_DVD_PKT_PES_SCRAMBLING
PES_packet contains scrambled data. This information message reports that the PES_scrambling_control is set to a value other than '0'.

>>> [DVD] ERROR 3216 (ref. DVD-3 Table 5.2.3-1 Note 1) :
ERR_DVD_PKT_NO_PDTS
PTS[32..0] and DTS[32..0] are mandatory in each Video PKT containing the first byte of the picture start code of any I-picture.

>>> [DVD] ERROR 3221 (ref. DVD-3 Table 5.2.3-1) :
ERR_DVD_VPKT_STRID
Video PES_packet stream_id shall be 1110 0000b

>>> [DVD] ERROR 3222 (ref. DVD-3 Table 5.2.3-1 Note 1) :
ERR_DVD_VPKT_PTS_NOSTRT
A Video packet shall not contain a PTS, if it does not contain the first byte of a picture start code.

>>> [DVD] ERROR 3224 (ref. DVD-3 Table 5.2.4-1 to 5.2.4-3) :
ERR_DVD_APKT_STRID
Audio PES_packet stream_id shall be:
- 1011 1101b (private_stream_1, Linear PCM, AC-3)
- 1100 0***b (packets containing MPEG1 audio, MPEG-2 audio without extension, MPEG-2 main audio with extension)
- 1101 0***b (packets containing MPEG-2 extension audio)

>>> [DVD] ERROR 3225 (ref. DVD-3 5.2.4) :
ERR_DVD_APKT_STR_NR
The Decoding Audio stream numbers shall not be assigned to the same number regardless of the audio compression mode.

>>> [DVD] ERROR 3226 (ref. DVD-3 Table 5.2.4-1 to 5.2.4-3 Note 1) :
ERR_DVD_APKT_PTS_LAST
The Last Audio PES_packet of a VOB shall have no PTS

>>> [DVD] ERROR 3227 (ref. DVD-3 Table 5.2.4-1 to 5.2.4-3 Note 1) :
ERR_DVD_APKT_PTS_GAP
An Audio PES_packet directly before an audio gap shall have no PTS

>>> [DVD] ERROR 3228 (ref. DVD-3 Table 5.2.4-3 Note 2) :
ERR_DVD_APKT_PTS_REST
The first A_PKT with the remainder of the previous A_PKT, should have no PTS

>>> [DVD] WARNING 3229 (ref. DVD-3 Table 5.2.4-1 Note 1) :
ERR_DVD_APKT_PTS_NOSTRT
An Audio packet shall not contain a PTS, if it does not contain the first sample of an audio frame.

>>> [DVD] ERROR 3231 (ref. DVD-3 Table 5.2.5-1 Note 1) :
ERR_DVD_SPKT_PTS_NOSTRT
A Sub Picture packet shall not contain a PTS, if it does not contain the first data of each Sub Picture Unit.

>>> [DVD] ERROR 3232 (ref. DVD-3 Table 5.2.5-1 Note 1) :
ERR_DVD_SPKT_PTS_EARLY
The earliest possible value of the PTS of the SPU is the arrival time of the last byte of the SPU in the Sub Picture Buffer.

>>> [DVD] ERROR 3235 (ref. DVD-3 Fig. 5.2.3/4/5-1) :
ERR_DVD_PKT_LEN_MAX
PES_packet length is 'number', resulting in a payload size of 'number'. This should be at most 'number' for stream packets. The maximum payload length for each packet is listed below:

packet type	**maximum payload length (bytes)**
video	2025
sub-picture	2025
LPCM	2017
AC3	2020
MPEG1 or MPEG-2 without extension	2020
MPEG-2 base	1152
MPEG-2 extension	1584

>>> [DVD] ERROR 3241 (ref. DVD-2 Table 5.2.1-1) :
ERR_DVD_PAD_PKT_LEN
The Padding packet length shall be at least 8.

9.3.6 DVD PES checks

>>> [DVD] ERROR 3251 (ref. DVD-3 Table 5.1-1) :
ERR_DVD_PES_STR_STRT
The beginning of each stream shall start from the first byte of each access unit.

>>> [DVD] ERROR 3252 (ref. DVD-3 Table 5.1-1) :
ERR_DVD_PES_STR_END
The end of each stream shall be aligned in each access unit. Therefore, when the pack length comprising the last data in each stream is less than 2048 bytes, it shall be adjusted by either method shown in DVD-3 Table 5.2.1-1.

>>> [DVD] ERROR 3261 (ref. DVD-3 5.4.1.3) :
ERR_DVD_VID_GAP_LEN
The interval between the presentation time of the picture which is stilled by the sequence_end_code and that of the next picture shall be equal or more than 0.4 seconds.

>>> [DVD] ERROR 3262 (ref. DVD-3 5.4.1.3 Restriction 1) :
ERR_DVD_VID_GAP_RATE
The Gap_length shall be an integer multiple of the video fields.

>>> [DVD] ERROR 3263 (ref. DVD-3 5.4.1.3 Restriction 2) :
ERR_DVD_VID_GAP_PARITY
If the Gap_length is a multiple of the video frame(= twice of video field), the last displayed field before the gap and the first displayed field after the gap shall have different field parities. In other cases they will have the same parity.

>>> [DVD] ERROR 3265 (ref. DVD-3 3.3.12.5 / 5.3) :

ERR_DVD_ESTD_UNDERFLOW
The ESTD buffer shall not underflow. (Data read from empty ESTD buffer.)

>>> [DVD] ERROR 3266 (ref. DVD-3 3.3.12.5 / 5.3) :
ERR_DVD_ESTD_OVERFLOW
The ESTD buffer shall not overflow. (Not enough space in ESTD buffer for data written to it.)

>>> [DVD] ERROR 3267 (ref. DVD-3 3.3.12.6) :
ERR_DVD_ESTD_ILL_INPUT
For the ESTD model: No packets shall arrive while STC - STC_offset is < 0.

>>> [DVD] ERROR 3268 (ref. DVD-3 3.3.12.6 / 5.3) :
ERR_DVD_ESTD_ILL_AUDIOGAP
For the ESTD model: Maximum two audio discontinuities (gaps) are allowed in a VOB.

>>> [DVD] ERROR 3269 (ref. DVD-3 3.3.12.6 / 5.3) :
ERR_DVD_ESTD_ILL_AUDIO_INP
For the ESTD model: o audio packets shall arrive while an audio gap is active.

>>> [DVD] ODDITY 3270 (ref. DVD-3 3.3.12.5 / 5.3) :
ERR_DVD_STD_NOT_EMPTY
For non seamless play the STD buffer is expected to be empty when decoding of a new VOB starts.

>>> [DVD] ERROR 3275 (ref. DVD-3 5.2.4.1 (2)) :
ERR_DVD_1ST_MPA_PKT_NONBASE
Audio stream does not start with a base stream packet (expected stream id ‘stream id’), but with a PES packet having a stream_id ‘stream id’

>>> [DVD] ERROR 3276 (ref. DVD-3 5.2.4.1 (2)) :
ERR_DVD_MPA_PKT_ALTER
Audio pack does not have an audio stream ‘extension or base’ stream packet (stream id ‘stream id’) followed by an ‘base or extension’ stream packet (expected stream id ‘stream id’), but by a PES packet having a stream_id ‘stream id’.

>>> [DVD] ERROR 3277 (ref. DVD-3 5.2.4.1 (2)) :
ERR_DVD_MPA_PKT_NRFRMS
The current audio packet (PES stream ‘stream id’) contains more than 1 audio ‘frame type’ frame !

>>> [DVD] ERROR 3278 (ref. DVD-3 5.2.4.1 (2)) :
ERR_DVD_MPA_ BASE_EXT_ORDER
Audio stream (PES stream_id ‘stream id’) base and extension frames should alternate : ‘frame type’ frame expected.

9.3.7 DVD Private stream checks

>>> [DVD] SYNTAX ERROR 3301 (ref. DVD-3 Table 5.1.1-2, 5.1.1-3) :
ERR_DVD_PRV_RES_SS_ID
sub_stream_id for private_stream_1

sub_stream_id	Stream coding
001* ****b	Sub-picture stream *****=Decoding Sub-picture stream number
0100 1000b	reserved
011* ****b	reserved (for extended Sub-picture)
1000 0***b	Dolby AC-3 audio stream ***=Decoding audio stream number
1000 1***b	DTS audio stream (option) ***=Decoding audio stream number
1001 0***b	SDDS audio stream (option) ***=Decoding audio stream number
1010 0***b	Linear PCM audio stream ***=Decoding audio stream number

1111 1111b	Provider defined stream
Others	reserved (for future Presentation Data)

sub_stream_id for private_stream_1

sub_stream_id	Stream coding
0000 0000b	PCI stream
0000 0001b	DSI stream
1111 1111b	Provider defined stream
Others	reserved (for future Navigation Data)

Note 1: "reserved" of sub_stream_id means that the sub_stream_id is reserved for future system extension. Therefore, it is prohibited to use reserved values of sub_stream_id.
Note 2: The sub_stream_id whose value is '1111 1111b' may be used for identifying a bitstream which is freely defined by the provider. However, it is not guaranteed that every player will have a feature to play that stream. The restriction of VOB, such as the maximum transfer rate of total streams, shall be applied, if the provider defined bitstream exists in VOB.

9.3.8 DVD Sequence header checks

>>> [DVD] ERROR 3351 (ref. DVD-3 Table 5.4.1.1-1, Table 5.4.1.2-1) :
ERR_DVD_HVSIZE_ILL
Sequence header horizontal_size x vertical_size shall be:

For MPEG-1 video:

TV system	525/60	625/50
horizontal_size x vertical_size	352x240	352x288

For MPEG-2 video:

TV system	525/60	625/50
horizontal_size x vertical_size	720 x 480	720 x 576
	704 x 480	704 x 576
	352 x 480	352 x 576
	352 x 240	352 x 288

>>> [DVD] ERROR 3352 (ref. DVD-3 5.4.1.1.1, 5.4.1.2.1) :
ERR_DVD_WIDTH_CH
Sequence and extension header horizontal_size shall be constant for all VOB's within a VOBS in a volume.

>>> [DVD] ERROR 3353 (ref. DVD-3 5.4.1.1.1, 5.4.1.2.1) :
ERR_DVD_HEIGHT_CH
Sequence and extension header vertical_size shall be constant for all VOBs within a VOBS in a volume.

>>> [DVD] ERROR 3354 (ref. DVD-3 Table 5.4.1.1-1, 5.4.1.2-1) :
ERR_DVD_ASP_RATIO_ILL
Sequence header aspect_ratio shall be:
- pel_aspect_ratio 4:3 for MPEG-1 video,
- Display aspect ratio 4:3 or 16:9 for MPEG-2 video.

>>> [DVD] ERROR 3355 (ref. DVD-3 Table 5.4.1.1-1, 5.4.1.2-1) :
ERR_DVD_FRM_RATE_ILL
Sequence header frame_rate_code shall be:

For MPEG-1 and MPEG-2 video:

TV system	525/60	625/50
frame rate	29.97 Hz	25 Hz

>>> [DVD] ERROR 3356 (ref. DVD-3 5.4.1.1.1, 5.4.1.2.1) :
ERR_DVD_FRM_RATE_CH
Sequence header frame_rate_code shall be identical for all VOBs within a VOBS in a volume.

>>> [DVD] ERROR 3357 (ref. DVD-3 Table 5.4.1.1-1) :
ERR_DVD_CP_FLAG_ILL
Sequence_header contrained_parameter_flag shall be 1

MPEG-1: Permitted combination of horizontal_size, vertical_size, frame_rate and aspect_ratio

horizontal_size	vertical_size	frame_rate	aspect_ratio
352	240	29.97	4:3
352	288	25	4:3

MPEG-2: Permitted combination of horizontal_size, vertical_size, frame_rate and display aspect_ratio

horizontal_size	vertical_size	frame_rate	aspect_ratio
720	480	29.97	16:9
720	480	29.97	4:3
704	480	29.97	16:9
704	480	29.97	4:3
352	480	29.97	4:3
352	240	29.97	4:3
720	576	25	16:9
720	576	25	4:3
704	576	25	16:9
704	576	25	4:3
352	576	25	4:3
352	288	25	4:3

>>> [DVD] ERROR 3358 (ref. DVD-3 Table 5.4.1.1-2, 5.4.1.2-2) :
ERR_DVD_FRM_VSIZE_ILL
Sequence_header : illegal vertical_size / frame_rate combination

>>> [DVD] ERROR 3359 (ref. DVD-3 Table 5.4.1.1-2, 5.4.1.2-2) :
ERR_DVD_HSIZE_VSIZE_ILL
Sequence_header : illegal horizontal_size / vertical_size combination

>>> [DVD] ERROR 3360 (ref. DVD-3 Table 5.4.1.1-2, 5.4.1.2-2) :
ERR_DVD_ASP_HSIZE_ILL
Sequence_header : illegal aspect_ratio / horizontal_size combination

>>> [DVD] ERROR 3361 (ref. DVD-3 Table 5.4.1.1-2, 5.4.1.2-2) :
ERR_DVD_ASP_PICR_ILL
Sequence_header : illegal aspect_ratio / frame_rate combination

>>> [DVD] ERROR 3365 (ref. DVD-3 5.4.1.2.1) :
ERR_DVD_PROFLEV_ILL
Sequence_extension profile_and_level_indication shall take the value 01001000b (MP@ML) or the value 01011000b (SP@ML).

>>> [DVD] ERROR 3366 (ref. DVD-3 Table 5.4.1.2-1) :
ERR_DVD_SEQ_BIT_RATE_LIM
Sequence_extension bitrate shall hold a constant value, for variable bitrate streams (vbv_delay coded as FFFFh) this shall be the maximum bitrate, it shall be equal or less than 9.80 Mbps.

>>> [DVD] ERROR 3367 (ref. DVD-3 Table 5.4.1.2-1) :

ERR_DVD_LOWDEL_1
Sequence_extension : low_delay should be 0

>>> [DVD] ERROR 3368 (ref. DVD-3 5.4.1.1-1 (*1)) :
ERR_DVD_BIT_RATE_ILL
Sequence_header : Bit_rate_field shall be 3FFFFh when the constrained_parameters_flag is set to 0.

>>> [DVD] ERROR 3370 (ref. DVD-3 5.4.1.2-2 (*2)) :
ERR_DVD_PROG_SEQ_ILL
Sequence_extension : progressive_sequence shall be 1 when the vertical_size equals 240

>>> [DVD] ERROR 3371 (ref. DVD-3 5.4.1.2.1) :
ERR_DVD_DISP_SIZE_ILL
Sequence_display_extension display_horizontal_size value shall be :

When aspect ratio is 16:9

horizontal_size	display_horizontal_size	aspect_ratio_information
720 or 704	720	16:9
720 or 704	540	4:3

When aspect ratio is 4:3

horizontal_size	display_horizontal_size	aspect_ratio_information
720 or 704	720	4:3
352	360	4:3

>>> [DVD] ERROR 3372 (ref. DVD-3 Table 5.4.1.2-5) :
ERR_DVD_DVSIZE_ILL
Sequence_display_extension display_vertical_size value shall be :

vertical_size	display_vertical_size
480	480
240	240
576	576
288	288

>>> [DVD] ERROR 3373 (ref. DVD-3 Table 5.4.1.2-5) :
ERR_DVD_VSIZE_DVSIZE_ILL
Sequence_display_extension : If vertical_size <> 480 or 240 or 576 or 288 then display_vertical_size shall be equal to vertical_size.

>>> [DVD] ERROR 3375 (ref. DVD-3 5.4.1.2 (6)) :
ERR_DVD_SEQEXT_DEF_ILL
Sequence display extension may or may not be present in the stream. The DVD-3 specification redefined the defaults for colour_primaries, transfer_characteristics and matrix_coefficients. If the frame rate is **25 Hz**:

- colour_primaries: The default value for this field shall be 5.
- transfer_characteristics: The default value for this field shall be 5.
- matrix_coefficients: The default value for this field shall be either 5 or 6 .

>>> [DVD] ERROR 3376 (ref. DVD-3 5.4.1.2 (6)) :
ERR_DVD_SEQEXT_DEF_ILL2
Sequence display extension may or may not be present in the stream. The DVD-3 specification redefined the defaults for colour_primaries, transfer_characteristics and matrix_coefficients. If the frame rate is **29.97 Hz**:

- colour_primaries: The default value for this field shall be either 4 or 6.
- transfer_characteristics: The default value for this field shall be either 4 or 6.
- matrix_coefficients: The default value for this field shall be either 5 or 6 .

9.3.9 DVD GOP checks

>>> [DVD] ERROR 3401 (ref. DVD-3 Table 5.4.1.1-1) :
ERR_DVD_NR_PICS_XS
Number of pictures in a GOP shall be:
- 18 display frames or less for TV system 525/60,
- 15 display frames or less for TV system 625/50.

>>> [DVD] ERROR 3402 (ref. DVD-3 5.4.1.4) :
ERR_DVD_NR_DISP_FLD_XS
The number of pictures in a GOP shall be equal to number_of_displayed_field_gop. For:
- **MPEG-2:**
  It shall be identical to the number of line21_data() recorded in the following loop,
- **MPEG-1, picture_rate 29.97Hz:**
  It shall be equal to the number of pictures multiplied by two.

>>> [DVD] ERROR 3403 (ref. DVD-3 5.4.1.4) :
ERR_DVD_TFFOG_PARITY_GAP
top_field_flag_of_gop shall not have a value such that there is a display field parity gap between previous other first GOPs in VOBUs.

>>> [DVD] ERROR 3404 (ref. DVD-3 5.4.1.4) :
ERR_DVD_USER_DATA_ILL
User data received without receiving Line 21 data.

>>> [DVD] ERROR 3405 (ref. DVD-3 5.4.1.4) :
ERR_DVD_USER_DATA_B_L21
User data received before Line 21 data.

>>> [DVD] ERROR 3406 (ref. DVD-3 5.4.1.4) :
ERR_DVD_L21_DATA_MISS
Line 21 data shall be present after every GOP header.

>>> [DVD] ERROR 3407 (ref. DVD-3 5.4.1.4) :
ERR_DVD_TFFOG_MISMATCH
Line 21 data top_field_flag_of_gop mismatch.

>>> [DVD] ERROR 3408 (ref. DVD-3 5.4.1.4) :
ERR_DVD_L21_DATA_ILL
No Line 21 data shall be recorded for the video gap caused by still pictures

>>> [DVD] ERROR 3409 (ref. DVD-3 5.4.1.4) :
ERR_DVD_NR_LINE21_DATA_XS
More that one user_data() for Line 21 data recorded in GOP.

>>> [DVD] ERROR 3410 (ref. DVD-3 5.4.1.4) :
ERR_DVD_SRSV_ILL
GOP User data for line 21, SRSV setting shall be set to 01F8h.

>>> [DVD] ERROR 3411 (ref. DVD-3 5.4.1.4) :
ERR_DVD_RES_ILL
Reserved_bit shall be 0.

>>> [DVD] ERROR 3412 (ref. DVD-3 5.4.1.4) :
ERR_DVD_MARKER_BITS_ILL
Marker_bits shall be all 1.

>>> [DVD] ERROR 3413 (ref. DVD-3 5.4.1.4) :
ERR_DVD_PARITY_ERR

- The MSB of line21_data1 is an odd parity bit which indicates the parity of the following 7-bits in line21_data1.
- The MSB of line21_data2 is an odd parity bit which indicates the parity of the following 7-bits in line21_data2.

9.3.10 DVD Picture checks

>>> [DVD] ERROR 3451 (ref. DVD-3 5.4.1.2.1) :
ERR_DVD_DISPM_FC_OFF_IL
frame_centre_horizontal_offset may only be different from 0 if the Display mode (in VMGM_V_ATR, VTSM_V_ATR or VTS_V_ATR) is 00b or 01b.

>>> [DVD] ERROR 3452 (ref. DVD-3 5.4.1.2.1) :
ERR_DVD_FRM_HOR_OFF

horizontal size	Permitted range (units 1/16th sample)
720	-1440 .. +1440
704	-1312 .. +1312

>>> [DVD] ERROR 3453 (ref. DVD-3 5.4.1.2.1) :
ERR_DVD_FRM_VER_OFF
frame_centre_vertical_offset shall always be 0.

>>> [DVD] ERROR 3460 (ref. DVD-3 5.4.1.2-1) :
ERR_DVD_VBV_DELAY_ILL
picture_header: vbv_delay value should be 0xFFFF for DVD.

9.3.11 DVD Audio checks

>>> [DVD] ERROR 3501 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_AFRM_ID
ID field shall not be set to lower sampling frequencies.

>>> [DVD] ERROR 3502 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_AFRM_LAYER
Layer shall be layer II.

>>> [DVD] ERROR 3503 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_PROT_BIT
protection_bit shall be zero.

>>> [DVD] ERROR 3504 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_AFRM_BITRATE

- Bitrate shall be between 64Kbps and 192Kbps for MPEG-1 and MPEG-2 main stream mono (1 channel).
- Bitrate shall be between 64Kbps and 384Kbps for MPEG-1 and MPEG-2 main stream stereo (2 channel).

>>> [DVD] ERROR 3505 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_AFRM_SAMPLFREQ
sampling_frequency shall be 48 kHz only.

>>> [DVD] ERROR 3506 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_PRIV_BIT
private_bit shall be zero.

>>> [DVD] ERROR 3507 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_AFRM_EMPH
Emphasis shall always be zero.

>>> [DVD] ERROR 3508 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_AFRM_MODE
Audio mode is 2 (dual_channel), which is not allowed in DVD.

>>> [DVD] ERROR 3511 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_SURROUND_MODE
Surround shall be:
- 00b, 01b or 10b for other than karaoke mode,
- 11b for karaoke mode.

>>> [DVD] ERROR 3512 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_DEMATRIX_PROCEDURE
Dematrix_procedure shall be 11b for unmatrixed mode, else always MPEG-1 compatible.

>>> [DVD] ERROR 3513 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_NMLCH
Number of multilingual channels shall be zero.

>>> [DVD] ERROR 3514 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_EXT_BIT_RATE
- Bitrate shall be up to 528 Kbps for MPEG-2 extension stream.
- Bitrate sum of main plus extension stream shall be up to 912Kbps.

>>> [DVD] ERROR 3515 (ref. DVD-3 5.4.2.3.1) :
ERR_DVD_NO_DRC
Not enough space for dynamic_range_control in audio base frame

>>> [DVD] ERROR 3516 (ref. DVD-3 Table 5.4.2.3-1) :
ERR_DVD_MC_PRED_ON
mc_prediction_on shall be zero.

>>> [DVD] ERROR 3517 (ref. DVD-3 Table 5.4.2.3.2.2) :
ERR_DVD_AUG_MTX_PROC_RES
aug_mtx_proc are two bits to indicate which dematrix procedure has to be applied for 7.1-channel audio signal. Values aug_mtx_proc==2 and aug_mtx_proc==3 are reserved.

>>> [DVD] ERROR 3518 (ref. DVD-3 Table 5.4.2.3.2.2) :
ERR_DVD_AUG_FTR_EXT_RES
aug_future_ext is for future extension, it should be zero

>>> [DVD] ERROR 3519 (ref. DVD-3 Table 5.4.2.3.2.2) :
ERR_DVD_DYNX_MODE7_FB
DynX_mode7 shall be at most 18.

>>> [DVD] ERROR 3520 (ref. DVD-3 5.4.2.3.1) :
ERR_DVD_DRC_RES_ILL
Reserved dynamic_range_control bits shall be '0'.

>>> [DVD] ERROR 3521 (ref. DVD-3 5.4.2.3.1) :
ERR_DVD_DRC_Y_ILL
Dynamic_range_control-Y component shall not exceed maximum 29.

9.3.12 DVD SPU checks

>>> [DVD] SYNTAX ERROR 3601 (ref. DVD-3 5.4.3.1 (1)) :
ERR_DVD_SPU_SIZE_0
SP unit size shall be > 0.

>>> [DVD] ERROR 3602 (ref. DVD-3 5.4.3.1 (1)) :
ERR_DVD_SPU_SIZE_ODD
SP unit size shall be even !

>>> [DVD] SYNTAX ERROR 3603 (ref. DVD-3 5.4.3.1 (1)) :
ERR_DVD_SPU_SIZE_ERR
SP unit size shall describe the size of a SPU in number of bytes.

>>> [DVD] ERROR 3604 (ref. DVD-3 5.4.3.1 (1)) :
ERR_DVD_SPU_SIZE
SP unit size shall be <= 53220 bytes.

>>> [DVD] ERROR 3605 (ref. DVD-3 5.4.3.1 (1)) :
ERR_DVD_SPU_DCSQT_SIZE
The size of SP_ DCSQT in a SPU shall be equal or less than half the size of the SPU.

>>> [DVD] SYNTAX ERROR 3611 (ref. DVD-3 5.4.3.1 (2)) :
ERR_DVD_SPU _DCSQTA_0
DCSQT_SA shall be > 0.

>>> [DVD] SYNTAX ERROR 3612 (ref. DVD-3 5.4.3.1 (2)) :
ERR_DVD_SPU _DCSQTA
SP_DCSQT_SA describes the start address of SP_DCSQT with RBN from the first byte of the SPU.

>>> [DVD] SYNTAX ERROR 3616 (ref. DVD-3 5.4.3.2 c)) :
ERR_DVD_SPU_PXD_XSIZE
Decoded PXD width shall be as set by SET_DAREA in SP_DCCMD.

>>> [DVD] ODDITY 3617 (ref. DVD-3 5.4.3.2 c)) :
ERR_DVD_SPU_PXD_XYSIZE
Decoded PXD size shall be as set by SET_DAREA in SP_DCCMD.

>>> [DVD] ERROR 3618 (ref. DVD-3 5.4.3.2 c)) :
ERR_DVD_SPU_PXD_XSIZE_MIS
PXD width of display area and bitmap pixel data shall be the same.
>>> [DVD] ERROR 3619 (ref. DVD-3 5.4.3) :
ERR_DVD_SPU_NO_PXD
PXD data shall be present.

>>> [DVD] SYNTAX ERROR 3621 (ref. DVD-3 5.4.3.3) :
ERR_DVD_SPU_DCSQT_0
SPU shall contain at least 1 DCSQ

>>> [DVD] ERROR 3622 (ref. DVD-3 5.4.3.3 (1)) :
ERR_DVD_SPU_DCSQ_STM_0
The SP_DCSQ_STM in the first SP_DCSQ shall be 0.

>>> [DVD] ERROR 3623 (ref. DVD-3 5.4.3.3) :
ERR_DVD_SPU_DCSQ_STM_ORD
SP_DCSQ i and SP_DCSQ j (i<j) shall be described in execution order (DCSQ_STM i < DCSQ_STM j).

>>> [DVD] ERROR 3624 (ref. DVD-3 5.4.3.3) :
ERR_DVD_SPU_DCSQ_STM_DUP
SP_DCSQ i and SP_DCSQ j (i<>j) shall not have the same execution start times.

>>> [DVD] ERROR 3625 (ref. DVD-3 5.4.3.3 (1)) :

ERR_DVD_SPU_DCSQ_STM_ILL
SP_DCSQ_STM shall be 0 or the positive integer value which is calculated by:
- (225 x n) / 64 (in case of TV system with 625/50)
- (3003 x n) / 1024 (in case of TV system with 525/60)

>>> [DVD] ERROR 3626 (ref. DVD-3 5.4.3.3 (2)) :
ERR_DVD_SPU_DCSQ_SA_NON
For tht last SP_DCSQ entry, DCSQ_SA shall point to itself.

>>> [DVD] ERROR 3627 (ref. DVD-3 5.4.3.3 (2)) :
ERR_DVD_SPU_DCSQ_SA_NXT
SP_ NXT_DCSQ_SA shall point to the RBN address of the next DCSQ entry.

>>> [DVD] SYNTAX ERROR 3628 (ref. DVD-3 5.4.3) :
ERR_DVD_SPU_DCSQ_PADD
Undefined data was found after the last SP_DCSQ.

>>> [DVD] ERROR 3631 (ref. DVD-3 5.4.3.3 (3)) :
ERR_DVD_SPU_DCCMD_0
SP_ DCSQ shall contain at least 1 DCCMD

>>> [DVD] ERROR 3632 (ref. DVD-3 5.4.3.3 (3)) :
ERR_DVD_SPU_DCCMD_DUP
The same SP_DCCMD shall not be described more than once.

>>> [DVD] ERROR 3634 (ref. DVD-3 Annex L) :
ERR_DVD_SPU_DCCMD_MIS
- Either FSTA_DSP or STA_DSP shall be described in the SP_DCSQ#0.
- SET_COLOR, SET_CONTR, SET_DAREA and SET_DSPXA shall be described in the SP_DCSQ#0.
- CMD_END shall be described.

>>> [DVD] ERROR 3635 (ref. DVD-3 Annex L) :
ERR_DVD_SPU_DCCMD_SIM
Two or more commands in FSTA_DSP, STA_DSP and STP_DSP shall not be described simultaneously in a SP_DCSQ.

>>> [DVD] ODDITY 3636 :
ERR_DVD_SPU_DCCMD_NO_PXD
A DCCMD was found in SP_DCSQ, but no PXD was found.

>>> [DVD] ERROR 3639 (ref. DVD-3 Annex L) :
ERR_DVD_SPU_DCCMD_END
The last DCCMD in a SP_DCSQ shall be CMD_END.

>>> [DVD] ERROR 3641 (ref. DVD-3 5.4.3.4 (6)) :
ERR_DVD_SPU_DCCMD_DAREA_CO
The origin of the Y-co-ordinate is SP line number 0. The origin of the X-co-ordinate is the starting point of the SP line number 0.
- X-co-ordinate values shall be in the range 0 to 719 inclusive.
- Y-co-ordinate values shall be in the range 2 to 479 inclusive for TV system 525/60.
- Y-co-ordinate values shall be in the range 2 to 574 inclusive for TV system 625/50.

>>> [DVD] ODDITY 3642 (ref. DVD-3 5.4.3.2) :
ERR_DVD_SPU_DCCMD_DSPXA_ORD
SET_DSPXA top field data address shall be lower than the bottom field address.

>>> [DVD] ERROR 3643 (ref. DVD-3 5.4.3.4 (7)) :

ERR_DVD_SPU_DCCMD_DSPXA_ADD
SET_DSPXA shall point to the first pixel of a run-length coded PXD line.

>>> [DVD] ERROR 3651 (ref. DVD-3 5.4.3.3 (1)) :
ERR_DVD_SPU_DCSQ_STM_PTS
Last DCSQ_STM shall be equal to or smaller than the PTS of the next SPU minus 1 video frame period.

>>> [DVD] ERROR 3652 (ref. DVD-3 5.1-1) :
ERR_DVD_SPU_DCSQ_STM_SVOB
The PTS of the first SPU is 'PTS value', but should be equal to or more than the VOB_V_S_PTM which is 'VOB_V_S_PTM value'.

>>> [DVD] ERROR 3653 (ref. DVD-3 5.1-1) :
ERR_DVD_SPU_DCSQ_STM_EVOB
The last PTM of the last SPU is 'PTM value' (PTS 'PTS value' + last_SP_DCSQ_STM*1024 'value' + video frame period 'value'), but should be equal to or less than the VOB_V_E_PTM which is 'VOB_V_E_PTM value'.

>>> [DVD] ERROR 3654 (ref. DVD-3 5.1-1) :
ERR_DVD_SPU_DCSQ_STM_ECEL
The last PTM of the last SPU is 'PTM value' (PTS 'PTS value' + last_SP_DCSQ_STM*1024 'value' + video frame period 'value'), but should be equal to or less than the Cell's presentation end time 'presentation time value'.

>>> [DVD] ERROR 3655 (ref. DVD-3 Annex L) :
ERR_DVD_SPU_DCCMD_ILL
For SP_DCSQ which is controlled by Highlight Information:
- CHG_COLCON shall not be described in any SP_DCSQ.
- STA_DSP shall not be described in any SP_DCSQ.
- SET_COLOR, SET_CONTR, SET_DAREA, SET_DSPXA and FSTA_DSP shall not be described in SP_DCSQ other than in the first SP_DCSQ.
- STP_DSP shall not be described in SP_DCSQ other than in the last SP_DCSQ.

>>> [DVD] ERROR 3656 (ref. DVD-3 Annex L) :
ERR_DVD_SPU_STA_DSP_INSYS
STA_DSP shall not be described in any SP_DCSQ in system space.

>>> [DVD] ERROR 3657 (ref. DVD-3 5.4.3.3 (6)) :
ERR_DVD_SPU_DCCMD_DAREA_YODD
SET_DAREA command Start Y-co-ordinate shall be even.

>>> [DVD] ERROR 3661 (ref. DVD-3 5.4.3.4.1 (1)) :
ERR_DVD_SPU_COLON_Y
CHG_COLCON command LN_CTLI, change line number shall be within:
- 2 .. 479 (TV system with 525/60)
- 2 .. 574 (TV system with 625/50)

>>> [DVD] ERROR 3662 (ref. DVD-3 5.4.3.4.1 (1)) :
ERR_DVD_SPU_COLCON_CHNR
CHG_COLCON command LN_CTLI Number_of_changes shall be within 1..8.

>>> [DVD] ERROR 3663 (ref. DVD-3 5.4.3.4.1 rule 1) :
ERR_DVD_SPU_COLCON_TERM
CHG_COLCON LN_CTLI Change termination line shall be greater or equal than the start line.

>>> [DVD] ERROR 3664 (ref. DVD-3 5.4.3.4.1 rule 2)) :

ERR_DVD_SPU_COLCON_STRT
CHG_COLCON LN_CTLI Change start line shall be larger than the previous LN_CTLI termination line.

>>> [DVD] ERROR 3665 (ref. DVD-3 5.4.3.4.1 rule 3)) :
ERR_DVD_SPU_COLCON_LN_ORD
CHG_COLCON LN_CTLI Change start line shall be in ascending order, thus larger than the previous LN_CTLI start lines.

>>> [DVD] ERROR 3666 (ref. DVD-3 5.4.3.4.1 rule 4)) :
ERR_DVD_SPU_COLCON_PX_ORD
CHG_COLCON LN_CTLI : In the group of PX_CTLI's immediately following each LN_CTLI, the change start pixel numbers in PX_CTLI shall be described in ascending order.

>>> [DVD] ERROR 3667 (ref. DVD-3 5.4.3.4.1 (2)) :
ERR_DVD_SPU_COLCON_PX_N8
CHG_COLCON LN_CTLI: At least 8 pixels with the same content shall be continued on the change start pixel and the pixels which follow.

>>> [DVD] SYNTAX ERROR 3671 (ref. DVD-3 5.4.3.1 (1)) :
ERR_DVD_SPU_PAD
SPU even size padding byte shall be 0xff .

>>> [DVD] ERROR 3672 (ref. DVD-3 5.4.3 Fig 5.4.3-2) :
ERR_DVD_SPU_PAD_PKT
A SP_PCK may have a padding packet, only when it is the last pack for a SPU.

>>> [DVD] ERROR 3673 (ref. DVD-3 5.4.3 Fig 5.4.3-2) :
ERR_DVD_SPU_STUFF
A SP_PCK packet may have stuffing bytes, only when it is the last packet for a SPU.

>>> [DVD] SYNTAX ERROR 3681 (ref. DVD-3 5.4.3.2 (a) 1-5) :
ERR_DVD_PXD_NPIX
- If 1 to 3 pixels with the same value follow, enter the number of the pixels followed in the first 2 bits and the pixel data in the following 2 bits. The 4 bits are considered to be one unit.
- If 4 to 15 pixels with the same value follow, specify 0 in the first 2 bits and enter the number of the pixels in the following 4 bits and the pixel data in the next 2 bits. The 8 bits are considered to be one unit.
- If 16 to 63 pixels with the same value follow, specify 0 in the first 4 bits and enter the number of the pixels in the following 6 bits and the pixel data in the next 2 bits. The 12 bits are considered to be one unit.
- If 64 to 255 pixels with the same value follow, specify 0 in the first 6 bits and enter the number of the pixels in the following 8 bits and the pixel data in the next 2 bits. The 16 bits are considered to be one unit.
- If the same pixels follow to the end of a line, specify 0 in the first 14 bits and describe the pixel data in the next 2 bits. The 16 bits are considered to be one unit.

>>> [DVD] SYNTAX ERROR 3685 (ref. DVD-3 5.4.3.2 (a) 7)) :
ERR_DVD_PXD_LINE_LONG
The size of the run-length coded data within one line shall be 1440 bits or less.

>>> [DVD] SYNTAX ERROR 3699 (ref. DVD-3) :
ERR_DVD_PRIV1_BITS_OVER
Packet parsing shall terminate with no bits left over !

9.3.13 AC-3 Checks

>>> [DVD] ERROR 3840 (ref. [AC-3] 7.10.2 (17)) :
ERR_DVD_AC3_PARSER_EXP_TOO_LARGE
While decoding packed exponents, the parser has encountered a packed exponent that is larger than 124, which is not allowed.

>>> [DVD] ERROR 3842 (ref. [AC-3] 5.4.3.11 + 5.4.3.12) :
ERR_DVD_AC3_PARSER_CPLBEGF_ERR
The parser detected that cplbegf is larger than cplendf + 2.
Since these values are used to calculate the number of coupling sub bands, the result will become negative, which is not allowed.

>>> [DVD] ERROR 3844 (ref. [AC-3] 5.4.4.1) :
ERR_DVD_AC3_PARSER_AUXDATA_NEG
The calculated size of the auxdata block resulted in a negative
value. Probably caused by an 'out of sync' problem in one of the previous audio blocks.

>>> [DVD] INFORMATION 3846 (ref. [AC-3] 5.4.1.1) :
ERR_DVD_AC3_PARSER_RECOVER
The parser encountered an error and skipped the remaining bits
of the current frame until the next syncword is encountered.

>>> [DVD] ERROR 3752 (ref. [AC-3] 5.4.1.1) :
ERR_DVD_AC3_SYNCWORD
The syncword should always be '0x0B77'.

>>> [DVD] ERROR 3754 (ref. [AC-3] 5.4.1.3) :
ERR_DVD_AC3_FSCOD_RESERVED
The fscod contains a reserved value.
Value ‘3’ is reserved.

>>> [DVD] ERROR 3756 (ref. [AC-3] 5.4.1.4) :
ERR_DVD_AC3_FRMSIZECOD_ILL
The frmsizecod should range between 0 and 18.

>>> [DVD] ERROR 3762 (ref. [AC-3] 5.4.2.4) :
ERR_DVD_AC3_CMIXLEV_RESERVED
The cmixlev contains a reserved value.
Value ‘3’ is reserved.

>>> [DVD] ERROR 3764 (ref. [AC-3] 5.4.2.5) :
ERR_DVD_AC3_SURMIXLEV_RESERVED
The surmixlev contains a reserved value.
Value ‘3’ is reserved.

>>> [DVD] ERROR 3766 (ref. [AC-3] 5.4.2.6) :
ERR_DVD_AC3_DSURMOD_RESERVED
The dsurmod contains a reserved value.
Value ‘3’ is reserved.

>>> [DVD] ERROR 3768 (ref. [AC-3] 5.4.2.15) :
ERR_DVD_AC3_ROOMTYP_RESERVED
The roomtyp contains a reserved value.
Value ‘3’ is reserved.

>>> [DVD] ERROR 3770 (ref. [AC-3] 5.4.2.23) :
ERR_DVD_AC3_ROOMTYP2_RESERVED
The roomtyp2 contains a reserved value.
Value ‘3’ is reserved.

>>> [DVD] ERROR 3772 (ref. [AC-3] 5.4.2.27) :
ERR_DVD_AC3_TIMECOD1_HRS_ILL
The time in hours (bits 1..5) of timecod1 should range between 0 and 23.

>>> [DVD] ERROR 3774 (ref. [AC-3] 5.4.2.27) :
ERR_DVD_AC3_TIMECOD1_MINS_ILL
The time in minutes (bits 6..11) of timecod1 should range between 0 and 59.

>>> [DVD] ERROR 3776 (ref. [AC-3] 5.4.2.28) :
ERR_DVD_AC3_TIMECOD2_FRMS_ILL
The time in frames (bits 4..8) of timecod2 should range between 0 and 29.

>>> [DVD] ERROR 3780 (ref. [AC-3] 7.10.2.(1)) :
ERR_DVD_AC3_CPLSTRE_ILL
The Coupling strategy (cplstre) should exist in the first audio block.

>>> [DVD] ERROR 3782 (ref. [AC-3] 7.10.2.(2)) :
ERR_DVD_AC3_NRCPLCHNS_IS_ZERO
The cplinu flag is set, but no channel is in coupling.

>>> [DVD] ERROR 3784 (ref. [AC-3] 7.10.2.(3)) :
ERR_DVD_AC3_CPLBEGF_TOO_LARGE
While cplinu is set, cplbegf should not be larger than cplendf+2.
However, when the parser detects that cplbegf is larger than cplendf, it is most likely that an error occurred. Because these fields are used in further bit allocation calculations, this error will lead to more serious parse errors. Therefore the parser will call the recover function to skip the current frame untill the next syncword is encountered.

>>> [DVD] ERROR 3786 (ref. [AC-3] 7.10.2.(4)) :
ERR_DVD_AC3_COUPLING_ILL
When a channel is in coupling, the Coupling coordinates should be transmitted in the first audio block, or the previous cplinu should be '0'.

>>> [DVD] ERROR 3788 (ref. [AC-3] 7.10.2.(5)) :
ERR_DVD_AC3_REMATSTR_ILL
No rematrix flags found in 2/0 audio.

>>> [DVD] ERROR 3790 (ref. [AC-3] 7.10.2.(6)) :
ERR_DVD_AC3_CPLEXPSTR_ILL
Coupling exponent strategy cannot specify 'reuse' in the first audio block or when the previous cplinu is '0'.

>>> [DVD] ERROR 3792 (ref. [AC-3] 7.10.2.(7)) :
ERR_DVD_AC3_CPLBEGF_DIFF
When coupling exponent strategy specifies 'reuse', cplbegf should equal the previous cplbegf.

>>> [DVD] ERROR 3794 (ref. [AC-3] 7.10.2.(7)) :
ERR_DVD_AC3_CPLENDF_DIFF
When coupling exponent strategy specifies 'reuse', cplendf should equal the previous cplendf.

>>> [DVD] ERROR 3796 (ref. [AC-3] 7.10.2.(8)) :
ERR_DVD_AC3_CHEXPSTR_ILL
The Channel exponent strategy cannot specify 'reuse' in the first audio block.

>>> [DVD] ERROR 3798 (ref. [AC-3] 7.10.2.(9)) :
ERR_DVD_AC3_CH_CPLBEGF_DIFF
When the channel exponent strategy specifies 'reuse', cplbegf should equal the previous cplbegf.

>>> [DVD] ERROR 3800 (ref. [AC-3] 7.10.2.(10)) :

ERR_DVD_AC3_LFEEXPSTR_ILL
The Lfe exponent strategy cannot specify 'reuse' in the first audio block when lfeon is set.

>>> [DVD] ERROR 3802 (ref. [AC-3] 7.10.2.(11)) :
ERR_DVD_AC3_CHBWCOD_TOO_LARGE
Chbwcod should range between '0-60'.

>>> [DVD] ERROR 3804 (ref. [AC-3] 7.10.2.(12)) :
ERR_DVD_AC3_BAIE_ILL
The 'bit allocation exists' (baie) should be set for the first audio block.

>>> [DVD] ERROR 3806 (ref. [AC-3] 7.10.2.(13)) :
ERR_DVD_AC3_SNROFFSTE_ILL
'SNR offset exists' (snroffste) should be set for the first audio block.

>>> [DVD] ERROR 3810 (ref. [AC-3] 7.10.2.(14)) :
ERR_DVD_AC3_CPLLEAKE_ILL
'Coupling leak initialization exists' (cplleake) should be set when cplinu is set for the first audio block.

>>> [DVD] ERROR 3830 (ref. DVD-3 Table 5.2.4-2 Note 3) :
ERR_DVD_AC3_NUM_FRAMEHEAD
The number_of_frame_headers should describe the number of audio frames whose first byte is in this A_PKT.

>>> [DVD] ERROR 3831 (ref. DVD-3 Table 5.2.4-2 Note 4) :
ERR_DVD_AC3_FIRST_AUPTR_ILL
The first_access_unit_pointer should describe the address of the first byte of the first AU in this A_PCK, with the RBN from the last byte of this field, but the first AU was found at RBN 'value'.

>>> [DVD] ERROR 3835 (ref. [AC-3] 7.10.1) :
ERR_DVD_AC3_PARSER_CRC_5_8_ERR
The CRC 'CRC value' at 5/8 of the frame should be 0. The frame is probably corrupt and might have, or could lead to other parser errors.

>>> [DVD] ERROR 3836 (ref. [AC-3] 7.10.1) :
ERR_DVD_AC3_PARSER_CRC_FULL_ERR
The CRC 'CRC value' at the end of the frame should be 0. The frame is probably corrupt and might have caused other parser errors.

>>> [DVD] ERROR 3837 (ref. [AC-3] 5.4.1.1) :
ERR_DVD_AC3_PARSER_INCOMPLETE_FRAME
Incomplete AC3 frame. The AC3 parser tried to parse more data than the length of the AC3 frame, indicated by the framecode. The AC3 parser will continue pasring the CRC error check at the end of the AC3 frame.

9.3.13.1 LPCM Private-1 Header Checks

The LPCM Private-1 Header checks are immediate checks. These checks are performed directly after the Private-1 header for LPCM packets is parsed.

>>> [DVD] ERROR 3851 (ref. DVD-3 Table 5.2.4-1) :
ERR_DVD_PR1H_RESERVED
Reserved fields in the Private-1 header of LPCM audio should describe '0'.

>>> [DVD] ERROR 3852 (ref. DVD-3 Table 5.2.4-1) :
ERR_DVD_PR1H_RESERVED_VALUE
The specified field in the Private-1 header of LPCM audio should describe a non-reserved value. This error is reported when:

- quantization_word_length specifies the value '11b'.
- audio_sampling_frequency specifies the value '10b' or '11b'.

>>> [DVD] ERROR 3853 (ref. DVD-3 Table 5.2.4-1 note 5) :
ERR_DVD_PR1H_EMPH_ILL
When the audio_sampling_frequency describes 96 kHz, the value of the audio_emphasis_flag should be '0b', describing 'emphasis off'.

>>> [DVD] ERROR 3854 (ref. DVD-3 Table 5.2.4-1 note 7) :
ERR_DVD_PR1H_FRM_NUM
The audio_frame_number should describe a number between '0' and '19'.

>>> [DVD] ODDITY 3855 (ref. DVD-3 Table 5.2.4-1) :
ERR_DVD_PR1H_FRM_NUM_ILL
The audio_frame_numbers should be assigned consecutively. This error reports that a non-consecutive audio_frame_number was found. This means that when an audio frame number was skipped or used twice this error is reported, but when only the current audio_frame_number is invalid, this error will be reported twice, as shown below.

	Audio frames:								
audio_frame_number	1	2	11	4	5	6	8	9	10
error generated			x	x			x		
correct value			3	!			7	(8)	(9)

As this constraint is not explicitly specified in the [DVD-3] specification, this error is reported as an ODDITY.

>>> [DVD] ERROR 3856 (ref. DVD-3 Table 5.2.4-1 note 7/5.4.2-1 (a)) :
ERR_DVD_PR1H_GOF_ILL
The audio_frame_number '0' should not be used when the GOF contains less then 20 audio frames, as audio_frame_number '0' is reserved for the beginning of a GOF and each GOF should consist of 20 audio frames(except for the last GOF in a VOB, which can consist of less then 20 audio frames).

This error reports that the verifier found a GOF that consists of less than 20 audio frames, but is not the last GOF of a VOB. This check is performed at the start of the next GOF and therefore will report a violation of the previous GOF.

>>> [DVD] ERROR 3857 (ref. DVD-3 Table 5.2.4-1 note 11) :
ERR_DVD_PR1H_DRC_YLARGE
The 'Y' component of the dynamic_range_control value, should describe a number between '0' and '29'. The 'Y' component of the dynamic_range_control value is defined as the 5 LSB bits (meaning a maximum value of 31) from the dynamic_range_control field in the Private-1 header for LPCM.

9.3.13.2 LPCM Audio Checks

The LPCM Audio checks are delayed checks. These checks are performed at the end of an Audio Pack.

>>> [DVD] ERROR 3870 (ref. DVD-3 Table 5.2.4-1 note 3) :
ERR_DVD_LPCM_NUM_FRMHEAD
The number_of_frame_headers should describe the number of audio frames whose first byte is in this A_PCK. This field, which is found in the Private-1 header for LPCM, is checked at the end of the A_PCK. Only at this moment can be determined if the specified number of audio frames started in the A_PCK.

>>> [DVD] ERROR 3871 (ref. DVD-3 Table 5.2.4-1 note 4) :
ERR_DVD_LPCM_FIRST_AUPTR_ILL
The first_access_unit_pointer should describe the address of the first byte of the first AU in this A_PCK, with the RBN from the last byte of this information. This field, which is found in the Private-1 header for LPCM, is checked at the end of the A_PCK. Only at this moment can be determined if the specified start address of the start of the first audio frame was correct.

>>> [DVD] ERROR 3872 (ref. DVD-3 Table 5.2.4-1 note 4) :
ERR_DVD_LPCM_FIRST_AUPTR_NOT_NULL
When the first byte of the first AU does not exist in this A_PCK, the first_access_unit_pointer should describe '0000 0000h'. This field, which is found in the Private-1 header for LPCM, is checked at the end of the A_PCK. Only at this moment can be determined if no audio frame started in the current A_PCK.

>>> [DVD] ERROR 3873 (ref. DVD-3 Table 5.2.4-1 note 7) :
ERR_DVD_LPCM_FRM_NUM_NA
The audio_frame_number should only specify '11111b', when no AU starts in the current A_PCK. This value is reserved for an audio frame when no first byte of any AU is present. This field, which is found in the Private-1 header for LPCM, is checked at the end of the A_PCK. Only at this moment can be determined if no audio frame started in the current A_PCK.

>>> [DVD] ERROR 3874 (ref. DVD-3 5.2.4 note 6) :
ERR_DVD_LPCM_MUTE_ON_DATA_ILL
When the flag audio_mute_flag specifies 'Mute On', all data from the A_PKT shall be zero. This check is not implemented because the validity is not clear.

>>> [DVD] ERROR 3875 (ref. DVD-3 5.4.2.1-2) :
ERR_DVD_LPCM_DATA_SIZE
Number_of_channels is too large in combination with the fs and quantisation (see [DVD-3] Table 5.4.2.1-2). The Number_of_channels should comply to this table. According to this table, there is a limit to the number of channels and bits per sample in combination with the sample frequency:

fs (kHz)	*quantization (bits)*	*16*	*20*	*24*
48		8	6	5
96		4	3	2

Table 1: Maximum number of audio_channels for any quantization/fs combination.

9.3.14 DVD VMG checks

>>> [DVD] ERROR 4001 (ref. DVD-3 4.1.1 / BP 12) :
ERR_DVD_VMG_EA_ILLEGAL
The value specified in the VMG_EA field contained an illegal value.

>>> [DVD] ERROR 4002 (ref. DVD-3 4.1.1 / BP 12) :
ERR_DVD_VMG_EA_SMALL
The value specified in the VMG_EA field was smaller than the value of the VMGI_EA field. Since the VMGI is a part of the VMG, this is illegal.

>>> [DVD] SYNTAX ERROR 4003 (ref. DVD-3 4.1.1) :
ERR_DVD_WRONG_OR_NO_VMGI
Input file error, probably caused by a non-VMGI stream being fed to the parser.

>>> [DVD] INFORMATION 4005 (ref. DVD-3 4.1.1 / BP 192) :
ERR_DVD_VOBS_FOUND
VMGI_MAT: Video Objects were found in the VMGI

>>> [DVD] INFORMATION 4007 (ref. DVD-3 4.1.1 / BP 192) :
ERR_DVD_VMGM_VOBS_SA_ILL
VMGI_MAT: Allocation mismatch: VMGM_VOBS_SA has value <value>
it must be <value>, according the file start locations in UDF file system.

>>> [DVD] ERROR 4010 (ref. DVD-3 4.1.1) :
ERR_DVD_RESERVED_BLOCK_ILL
A number of bits in a block of reserved fields were non-zero.

These blocks are not parsed, but only skipped and checked for a non-zero value. These blocks of reserved fields are generally more than 4 bytes long and as such identified in the specification.

>>> [DVD] ERROR 4011 (ref. DVD-3 2.1) :
ERR_DVD_RESERVED_FIELD_ILL
All reserved fields should have all their bits cleared.

>>> [DVD] ERROR 4012 (ref. DVD-3 4.1.1) :
ERR_DVD_RESERVED_VALUE_ILL
A field specified a value that is reserved (i.e. not to be used).

>>> [DVD] ERROR 4013 (ref. DVD-3 4.1.1 / BP 0) :
ERR_DVD_VMG_ID_INVALID
The VMG_ID field should describe 'DVDVIDEO-VMG' to identify the VMGI file. Only characters from the ISO646 (a-characters) are allowed. This message indicates that this field did not specify the correct string and could indicate an input-stream other than a VMGI.

>>> [DVD] ERROR 4014 (ref. DVD-3 4.1.1 / BP 32) :
ERR_DVD_VERN_INVALID
The VERN (Version number of DVD Video Specifications) field should specify '00010000b', other values are prohibited.

>>> [DVD] ERROR 4015 (ref. DVD-3 4.1.1) :
ERR_DVD_NS_TOO_SMALL
A number field was found that specified the number '0'. This message refers to all the number fields that have a legal range starting from '1', e.g. VTS_Ns, VMGM_AST_Ns, etc.

>>> [DVD] ERROR 4016 (ref. DVD-3 4.1.1) :
ERR_DVD_NS_TOO_LARGE
A number field was found that specified a number larger than '99'. This message refers to all the number fields that have a maximum value '99', e.g. VTS_Ns, VMGM_AST_Ns, etc.

>>> [DVD] ERROR 4017 (ref. VCPS G.1.2) :
ERR_DVD_VMGI_VCPS_ID_INVALID
VMGI_MAT: The VCPS_ID is (<value>) and should be 'VCPS'.

>>> [DVD] ERROR 4020 (ref. DVD-3 4.1.1 / BP 64) :
ERR_DVD_ID_NON_A_CHARACTER
This error is reported for the VMG_ID and the PVR_ID fields. It indicates that some characters from the field were not from the a-characters set according to the ISO646 charactercode table. The definition of these a-characters can be found in [DVD-2] Chapter 1.5.3.

>>> [DVD] ERROR 4021 (ref. DVD-3 4.1.1 / BP 0/ BP 64) :
ERR_DVD_ID_ISO646_ILLEGAL
This error is reported for the VMG_ID and the PVR_ID fields. It indicates that some characters from the field were illegal according to the ISO646 charactercode table, i.e. characters with charactercode exceeding '127'.

>>> [DVD] ERROR 4022 (ref. DVD-3 4.1.1 /BP 0/ BP 64) :
ERR_DVD_ID_NON_PRINTABLE
This error is reported for the VMG_ID and the PVR_ID fields. It indicates that some characters from the field were non-printable, e.g. Linefeed, Backspace, Carriage return, etc.

>>> [DVD] ODDITY 4023 (ref. DVD-3 4.1.1 / BP 64) :
ERR_DVD_ID_NULL
This checks if the PVR_ID field contains data. If all the bytes from this field are '0', this ODDITY will be generated.

>>> [DVD] ERROR 4025 (ref. DVD-3 4.1.1 / BP 128) :
ERR_DVD_VMGI_MAT_TOO_LARGE
The VMGI_MAT_EA was too large. The maximum size of the VMGI_MAT is '2291' ('08F3h') bytes. Therefore, the maximum value of the VMGI_MAT_EA field is '2290' ('08F2h').

>>> [DVD] ERROR 4026 (ref. DVD-3 4.1.1 / BP 128) :
ERR_DVD_VMGI_MAT_TOO_SMALL
The VMGI_MAT_EA was too small. The minimum size of the VMGI_MAT is '1024' ('0400h') bytes. Therefore, the minimum value of the VMGI_MAT_EA field is '1023' ('03FFh').

>>> [DVD] ERROR 4027 (ref. DVD-3 4.1.1 / BP 128) :
ERR_DVD_VMGI_MAT_ILL
The VMGI_MAT_EA should be '1023' ('03FFh') when no FP_PGCI exists in the VMGI_MAT.

>>> [DVD] ERROR 4028 (ref. DVD-3 4.1.1 / BP 132) :
ERR_DVD_FP_PGCI_SA_ILL
The FP_PGCI_SA should be '1024' ('0400h') when a FP_PGCI exists in the VMGI_MAT. No other start address may be specified.

>>> [DVD] ERROR 4029 (ref. DVD-3 4.1.1 / BP 132) :
ERR_DVD_VMGI_NO_FP_PGCI
No FP_PGCI specified in the VMGI_MAT of this VMGI. This message is reported as an INFORMATION message and can be useful in tracking problems down.

>>> [DVD] ERROR 4031 (ref. DVD-3 4.1.1 / BP196) :
ERR_DVD_NO_TT_SRPT
The TT_SRPT (Title Search Pointer Table) is mandatory, but missing in this VMGI (the TT_SRPT_SA field was zero).

>>> [DVD] ERROR 4032 (ref. DVD-3 4.1.1 / BP 200) :
ERR_DVD_NO_VMGM_PGCI_UT_SA (BP 200)
When VOBS are associated with the VMGI, indicated by a non-zero value of VMGM_VOBS_SA, the VMGM_PGCI_UT (Video Manager Menu PGCI Unit Table) is mandatory, but missing in this VMGI (the VMGM_PGCI_UT_SA field was zero).

>>> [DVD] ERROR 4035 (ref. DVD-3 4.1.1 / BP 216) :
ERR_DVD_NO_C_ADT_SA
When VOBS are associated with the VMGI, indicated by a non-zero value of VMGM_VOBS_SA, the VMGM_C_ADT (Video Manager Menu Cell Address Table) is mandatory, but missing in this VMGI (the VMGM_C_ADT_SA field was zero).

>>> [DVD] ERROR 4036 (ref. DVD-3 4.1.1 / BP 220) :
ERR_DVD_NO_VMGM_VOBU_ADMAP_SA
When VOBS are associated with the VMGI, indicated by a non-zero value of VMGM_VOBS_SA, the VMGM_VOBU_ADMAP (Video Manager Menu Video Object Unit Address Map Table) is mandatory, but missing in this VMGI (the VMGM_VOBU_ADMAP_SA field was zero).

>>> [DVD] ERROR 4037 (ref. DVD-3 4.1.1 / BP 258) :
ERR_DVD_VMGM_AST_NS_TOO_LARGE
There can be only one Audio stream (VMGM_AST_Ns) associated with the VMGI VOBS. This error reports there was more than one audio stream specified.

>>> [DVD] ERROR 4038 (ref. DVD-3 4.1.1 / BP 258) :
ERR_DVD_VMGM_AST_NS_NOT_NULL
When no VOBS are associated with the VMGI, indicated by a zero value of VMGM_VOBS_SA, the number of Audio streams (VMGM_AST_Ns) should be '0'.

>>> [DVD] ERROR 4039 (ref. DVD-3 4.1.1 / BP 258) :

ERR_DVD_VMGM_SPST_NS_TOO_LARGE

There can be only one Sub-picture stream (VMGM_SPST_Ns) associated with the VMGI VOBS. This error reports there is more than one Sub-picture stream specified.

>>> [DVD] ERROR 4040 (ref. DVD-3 4.1.1 / BP 258) :
ERR_DVD_VMGM_SPST_NS_NOT_NULL

When no VOBS are associated with the VMGI, indicated by a zero value of VMGM_VOBS_SA, the number of Sub-picture streams (VMGM_SPST_Ns) should be '0'.

>>> [DVD] ERROR 4041 (ref. DVD-3 4.1.1) :
ERR_DVD_DATA_FOUND_WITHOUT_VOBS

When no VOBS are associated with the VMGI, indicated by a zero value of VMGM_VOBS_SA, the Video, Audio and Sub-picture stream attributes should specify '0' in every bit. This error indicates some bits from an attribute field were non-zero.

>>> [DVD] ERROR 4042 (ref. DVD-3 4.1.1) :
ERR_DVD_TABLE_FOUND_WITHOUT_VOBS

A table was found (i.e. the start address field contained a non-zero value) in the VMGI, that is only required when VOBS are associated with this VMGI. When no VOBS are associated, these tables should not exist.

This error can specify the following tables:

- VMGM_PGCI_UT
- VMGM_C_ADT
- VMGM_VOBU_ADMAP

>>> [DVD] ERROR 4045 (ref. DVD-3 4.1.1) :
ERR_DVD_EA_SMALL

An end address field specified a value that is too small, usually '0000h'.

>>> [DVD] ERROR 4046 (ref. DVD-3 4.1.1) :
ERR_DVD_EA_EQUAL_SA

In lists where both the start address and the end address of a particular block are given, e.g. in the VMGM_C_ADT (Video Manager Menu Cell Address Table), the end addresses of the (N)th block should not be equal to the start addresses of the (N+1)th block. If this is the case, this error is generated, stating that the end address of the (N)th block should be the Start Address of the (N+1)th block minus '1'.

This error might also indicate that the start address of the (N+1)th block is invalid.

>>> [DVD] ERROR 4047 (ref. DVD-3 4.1.1) :
ERR_DVD_SA_ILLEGAL

A start address field specified the value '0000h', which is not valid, because this value would indicate that a block of data would start at the beginning of the current Logic Block.

>>> [DVD] ERROR 4048 (ref. DVD-3 4.1.1) :
ERR_DVD_SA_ILL_ORDER

In lists where start addresses of blocks are specified, these addresses should be specified in ascending order. When the start address of the (N)th block is smaller than the start address of the (N-1)th block, this error is generated.

This error can specify the following tables:

- VMGM_PGCI_UT
- PTL_MAIT
- VMGM_C_ADT

>>> [DVD] ERROR 4049 (ref. DVD-3 4.1.1) :
ERR_DVD_EA_PAST_SA

In lists where both the start address and the end address of a particular block are given, e.g. in the VMGM_C_ADT (Video Manager Menu Cell Address Table), the end addresses of the (N)th block should be smaller than the start addresses of the (N+1)th block, as this would indicate overlapping datablocks.

This error might also indicate that the start address of the (N+1)th block is invalid.

>>> [DVD] ERROR 4050 (ref. DVD-3 4.1.1) :
ERR_DVD_TABLE_POS_ERR
The start address of a table is not equal to the start address specified in the VMGI_MAT.

This error can specify the following tables:

- TT_SRPT
- VMGM_PGCI_UT
- PTL_MAIT
- VTS_ATRT
- TXTDT_MG
- VMGM_C_ADT
- VMGM_VOBU_ADMAP

>>> [DVD] ERROR 4051 (ref. DVD-3 4.1.1) :
ERR_DVD_SRP_ILL
This error is reported when a VMGM_PGCI_SRP does not point to the correct address. The closest address is also reported, to provide some help while solving this error.

>>> [DVD] ERROR 4052 (ref. DVD-3 4.1.1) :
ERR_DVD_SRP_SA_INVALID
This error is reported when a Search pointer does not point to the correct address. The correct address is specified. This error can be reported in:

- VMGM_LU_SRP table
- PLT_MAI_SRP table
- VTS_ATR_SRP table
- TXTDT_LU_SRP table
- IT_TXT_SRP table

>>> [DVD] ERROR 4055 (ref. DVD-3 4.1.1 / BP 34) :
ERR_DVD_RMA_NOT_DEFINED
At least one Region Management (RMA8….RMA1) field in the VMG_CAT should be '0', indicating a region where this disc is allowed to be played. When all the Region Management fields are '1', the disc cannot be played in any region, making this disc useless. As this condition is not specified as an error in the [DVD-3] specification, it will be reported as an oddity by the DVD-Video verifier.

>>> [DVD] ERROR 4056 (ref. DVD-3 4.1.1 / BP 38) :
ERR_DVD_NUM_VOLUMES_ZERO
The Number_of_Volumes field in VMLS_ID must be at least '1'. This error will be generated if the Number_of_Volumes equals '0'.

>>> [DVD] ERROR 4057 (ref. DVD-3 4.1.1 / BP 38) :
ERR_DVD_VOLUME_NUMBER_NULL
The Volume_number field in VMLS_ID must be at least '1'. This error will be generated if the Volume_number equals '0'.

>>> [DVD] ERROR 4058 (ref. DVD-3 4.1.1 / BP 38) :
ERR_DVD_VOLUME_NUMBER_TOO_BIG
The Volume_number field in VMLS_ID can be maximum the Number_of_Volumes, defined in VMLS_ID. This error will be generated when the Volume_number exceeds the Number_of_Volumes.

>>> [DVD] ERROR 4060 (ref. DVD-3 4.1.1 / BP 256) :

ERR_DVD_SOURCE_PIC_RES_RESERVED

A value from the range '100b'…'111b' was specified for the Source_picture_resolution field in VMGM_V_ATR. These are reserved values and should not be specified, only values from '000b'...'011b' can be used.

>>> [DVD] ERROR 4061 (ref. DVD-3 4.1.1 / BP 256) :
ERR_DVD_SOURCE_PIC_LETTBOX_ILL

The Source_picture_letterboxed field should be set to '0' for the 16:9 Aspect_ratio.

>>> [DVD] ERROR 4062 (ref. DVD-3 4.1.1 / BP 256) :
ERR_DVD_DISPLAY_MODE_ILL

This error reports that the specified Display_mode is illegal for the specified Aspect_ratio. The error occurs when:

- Display_mode equals '00b'…'10b' when the Aspect_ratio equals '00b' (4:3).
- Display_mode equals '11b' when the Aspect_ratio equals '11b' (16:9).

>>> [DVD] ERROR 4064 (ref. DVD-3 4.1.1 / BP 260s) :
ERR_DVD_AUDIO_CODING_MODE_ILL

This error reports that the specified Audio_coding_mode is illegal for the specified TV_system from the VMGM_V_ATR. The error occurs when:

- Audio_coding_mode equals '010b' (MPEG-1) or '011b' (MPEG-2), when the TV_system equals '00b' (NTSC).
- Audio_coding_mode equals '000b' (AC-3) when the TV_system equals '01b' (PAL).

>>> [DVD] ERROR 4065 (ref. DVD-3 4.1.1 / BP 260s) :
ERR_DVD_TV_SYSTEM_ILL

This error occurs when the TV_system specified in the script is different from the TV_system field found in the VMGM_V_ATR. The TV_system from the script-file is used for verification of the VMGI, errors regarding the TV_system, Source_picture_resolution, tc_flag and audio_coding_mode can be caused by this.

>>> [DVD] ERROR 4066 (ref. DVD-3 4.1.1 / BP 260) :
ERR_DVD_QUANTIZATION_RESERVED

This error reports that the specified Quantization/DRC field is illegal for the specified Audio_coding_mode. The error occurs when:

- Quantization/DRC field does not equal '11b', when the Audio_coding_mode equals '000b' (Dolby AC-3).
- Quantization/DRC field equals '11b', when the Audio_coding_mode equals '100b' (LPCM).

>>> [DVD] ERROR 4067 (ref. DVD-3 4.1.1 / BP 260) :
ERR_DVD_DRC_RESERVED

This error reports that the specified Quantization/DRC field is illegal for the specified Audio_coding_mode. The error occurs when:

- Quantization/DRC field equals '10b' or '11b', when the Audio_coding_mode equals '010b' (MPEG-1) or '011b' (MPEG-2).

>>> [DVD] ERROR 4068 (ref. DVD-3 4.1.1 / BP 260) :
ERR_DVD_FS_RESERVED

The Frequency (fs) field specified the reserved value '10b' or '11b'.

>>> [DVD] ERROR 4069 (ref. DVD-3 4.1.1 / BP 260) :
ERR_DVD_FS_ILL_MPEG
This error reports that the Frequency (fs) field specified an illegal sampling frequency for the specified Audio_coding_mode. This error occurs when:

- The Audio_coding_mode does not equal '100b' (LPCM) and the Frequency (fs) field specifies the value '01b' (96 kHz).

>>> [DVD] ERROR 4070 (ref. DVD-3 4.1.1 / BP 260) :
ERR_DVD_AUDIO_CHANN_TOO_LARGE
The Number_of_audio_channels must not be larger than the defined Number of audio channels for the specified Audio_coding_mode. This error occurs when:

- The Number_of_audio_channels is larger than '001b' (2ch), when the Audio_coding_mode equals '100b' (LPCM).
- The Number_of_audio_channels is larger than '001b' (2ch), when the Audio_coding_mode equals '010b' (MPEG-1).
- The Number_of_audio_channels is larger than '101b' (5ch + 0.1ch), when the Audio_coding_mode equals '000b' (AC-3).

>>> [DVD] ERROR 4075 (ref. DVD-3 4.1.1 / BP 342) :
ERR_DVD_SUBPIC_CODING_MODE_RES
The Sub-picture_coding_mode field specified a reserved value.

>>> [DVD] ERROR 4080 (ref. DVD-3 4.1.2 / (2)) :
ERR_DVD_TT_SRPT_AGL_TOO_LARGE
The AGL_Ns field exceeds '9'. A maximum of '9' Angles are allowed.

>>> [DVD] ERROR 4081 (ref. DVD-3 4.1) :
ERR_DVD_NS_NUM_TOO_SMALL

>>> [DVD] ERROR 4082 (ref. DVD-3 4.1.1) :
ERR_DVD_NS_NUM_TOO_LARGE

>>> [DVD] ERROR 4083 (ref. DVD-3 4.1.2-2 / (5)) :
ERR_DVD_TT_VTSN_ILL

>>> [DVD] ERROR 4084 (ref. DVD-3 4.1.2-2 / (5)) :
ERR_DVD_TT_TTN_NON_CONT
This error is reported when the VTS_TTN was found not to be continuous. For each VTSN in the TT_SRP, the VTS_TTN must be assigned continuously, i.e. no gaps are allowed in the assignment of the VTS_TTN.

>>> [DVD] ERROR 4085 (ref. DVD-3 4.1.2-2 / (5)) :
ERR_DVD_TT_TTN_NOT_INC
This error is reported when the VTS_TTN was found to have the same value as the VTS_TTN in the previous TT_SRP for the same VTSN. For each VTSN in the TT_SRP, the VTS_TTN must be assigned continuously, i.e. no identical VTS_TTN values are allowed. This check is not implemented because it was found during creation of this document.

>>> [DVD] ERROR 4086 (ref. DVD-3 4.1.2-2 / (3)) :
ERR_DVD_PTT_NS_TOO_LARGE
This error is reported when the PTT_Ns field contains a value that is too large. This could indicate an error in the value of the TT_TY. Valid values for the PTT_Ns are:

- The PTT_Ns value must be in the range '1' to '99' when the TT_TY is '0b' (One_sequential_PGC_title).
- The PTT_Ns value must be in the range '1' to '999' when the TT_TY is '1b' (One_random_PGC_title or Multi_PGC_title).

>>> [DVD] ERROR 4087 (ref. DVD-3 4.1.2-1 / (2)) :
ERR_DVD_TT_SRPT_EA_INVALID
The end address specified in the TT_SRPT_EA field is not correct. The value that is most likely to be the correct value is reported by this error. This could also indicate an error in the TT_SRP_Ns value.

>>> [DVD] ERROR 4090 (ref. DVD-3 4.1) :
ERR_DVD_TXTDT_SRP_SURP

>>> [DVD] ERROR 4091 (ref. DVD-3 4.1.6-3) :
ERR_DVD_TXTDT_SRP_ILL

>>> [DVD] ERROR 4092 (ref. DVD-3 4.1.3-2 / (1)) :
ERR_DVD_VMGM_LCD_REUSED
This error is reported when a language code was used more than once. A language shall only appear once in the table. This error can specify the following tables:

- VMGM_LU_SRP
- TXTDT_LU_SRP

>>> [DVD] ERROR 4093 (ref. DVD-3 4.1.3-2 / (1)) :
ERR_DVD_VMGM_LCD_ILL
This error is reported when a language code value is not valid. Valid language code values are found in the [DVD-3] Annex B. This error can specify the following tables:

- VMGM_LU_SRP
- TXTDT_LU_SRP

>>> [DVD] ERROR 4095 (ref. DVD-3 4.1.3.1-2 / (1)) :
ERR_DVD_NO_ENTRY_MENU
At least one VMGM_PGCI_SRP in the VMGM_LU must have the Menu_ID field in VMGM_PGC_CAT set to '0010b' (Title Menu), when the TTM_EXST in VMGM_EXST is set to '1b'. This error is reported when this is not the case, indicating that no Entry menu is present for the PGCI this PGC_SRP points to, which is not allowed.

>>> [DVD] ERROR 4096 (ref. DVD-3 4.1.3.1-2 / (1)) :
ERR_DVD_MENU_NOT_FOUND

>>> [DVD] ERROR 4097 (ref. DVD-3 4.1.3-2 / (1)) :
ERR_DVD_MORE_ENTRY_MENU
Only one Title Menu is allowed for each VMGM_LU. This error will be reported when more than one Title Menu was found.

>>> [DVD] ERROR 4098 (ref. DVD-3 4.1.3-2 / (1)) :
ERR_DVD_MENU_ID_ILL
The Menu_ID value was illegal in combination with the Entry_type (in VMGM_PGC_CAT). This error is reported when:

- The Menu_ID does not equal '0000b' when the Entry_type equals '0b' (Not Entry PGC).
- The Menu_ID does not equal '0010b' when the Entry_type equals '1b' (Entry PGC).

>>> [DVD] ERROR 4099 (ref. DVD-3 4.1.3-2 / (1)) :
ERR_DVD_BLOCK_MODE_ILL
The Block_mode value was illegal in combination with the Block_type (in VMGM_PGC_CAT). This error will be reported when:

- The Block_mode does not equal '00b' (Not a PGC in the Block) when the Block_type equals '00b' (Not part of a block).

>>> [DVD] ERROR 4100 (ref. DVD-3 3.3.3) :
ERR_DVD_BLOCK_MODE_ILL2
The Block_mode value was illegal in combination with the Block_type (in VMGM_PGC_CAT). This error will be reported when:

- The Block_mode equals '00b' when the Block_type equals '01b' (Parental block).

>>> [DVD] ODDITY 4105 (ref. DVD-3 4.1.4-1 / BP 1) :
ERR_DVD_CTY_NS_ILL
The CTY_Ns should be in the range '1'…'255'. This error reports an illegal value.

>>> [DVD] ERROR 4106 (ref. DVD-3 4.1.4-1 / BP 1) :
ERR_DVD_VTS_NS_ERR
The VTS_Ns specified in the PTL_MAITI must have the same value as the VTS_Ns specified in the VMGI_MAT.

>>> [DVD] ERROR 4107 (ref. DVD 4.1.3-1 / (2)) :
ERR_DVD_EA_ERROR
When the complete table is parsed, the current position is compared with the specified end address of the table (e.g. The PTL_MAIT_EA field). When these values do not match, this error is reported, indicating that the end address field value was wrong.

This error can also indicate a problem with a number field (e.g. The VMGM_PGCI_SRP_Ns field) or a flag, which causes the wrong number of fields to be parsed.

>>> [DVD] ERROR 4110 (ref. DVD-3 4.1.4-2 / (BP 1)) :
ERR_DVD_VMGM_CTY_CD_REUSED
A CTY_CD (Country code) was used more than once. A Country code shall only appear once in the PTL_MAI_SRP table.

>>> [DVD] ERROR 4111 (ref. DVD 4.1.4-2 / (BP 1)) :
ERR_DVD_VMGM_CTY_CD_ILL
A CTY_CD (Country code) value is not valid. Valid CTY_CD values are found in the ISO-3166 Alpha-2 specification.

>>> [DVD] ERROR 4115 (ref. DVD-3 4.1.4.1 / BP 258) :
ERR_DVD_PTL_ID_VMG_NO_VOBS
When no VOBS are associated with the VMGI, indicated by a zero value of VMGM_VOBS_SA, the PTL_ID_VMG should specify the value '0000h'.

>>> [DVD] ERROR 4116 (ref. DVD-3 4.1.6-2 / (2)) :
ERR_DVD_TXTDT_CHRS_UNICODE
The CHRS (Character set) field in TXTDT_LU_SRP specifies the value '00h', which is reserved for Unicode.

>>> [DVD] ERROR 4120 (ref. DVD-3 4.1.7-2 / (1)) :
ERR_DVD_VOB_IDN_ORDER
The VMGM_VOB_IDN field from VMGM_CPI should be assigned continuously. This error will be reported when a gap was found in the assignment of the VMGM_VOB_IDN values.

>>> [DVD] ERROR 4121 (ref. DVD-3 4.1.7-2 / (1)) :

ERR_DVD_VOB_IDN_ILL
The VMGM_VOB_IDN value exceeds the number of VOBs in the VMGM_VOBS, specified by the VMGM_VOB_Ns field in VMGM_C_ADTI.

>>> [DVD] ERROR 4122 (ref. DVD-3 4.1.7-2 / (1)) :
ERR_DVD_NEW_VOB_C_IDN_ILL
The first VMGM_C_IDN value from VMGM_CPI for each VMGM_VOB_IDN does not equal '1'.

>>> [DVD] ERROR 4123 (ref. DVD-3 4.1.7-2 / (2)) :
ERR_DVD_C_IDN_ORDER
The VMGM_C_IDN field from VMGM_CPI should be assigned continuously for each VMGM_VOD_IDN. This error will be reported when a gap was found in the assignment of the VMGM_C_IDN values.

>>> [DVD] ERROR 4130 (ref. DVD-3 4.1.8-1 / (1)) :
ERR_DVD_ADMAP_EA_OVER
The VMGM_VOBU_ADMAP_EA value from the VMGM_VOBU_ADMAPI is not correct. This error is reported when the parser cannot read another VMGM_VOBU_AD from the file without reading beyond the VMGM_VOBU_ADMAP_EA. This error is reported as a System Error.

>>> [DVD] ERROR 4131 (ref. DVD-3 4.1.7-1 / (3)) :
ERR_DVD_VMGM_CP_SA_IN_USE
The VMGM_CP_SA field from VMGM_CPI was already specified by another Cell piece.

>>> [DVD] ERROR 4132 (ref. DVD-3 4.1.8-1) :
ERR_DVD_VOBU_NS_ILL
The VMGM_VOB_Ns value from the VMGM_VOBU_ADMAPI does not correspond with the number of VOBU addresses read from the file. This could indicate a problem with the VMGM_VOBU_ADMAP_EA field.

>>> [DVD] ERROR 4133 (ref. DVD-3 4.1.8-1 / (1)) :
ERR_DVD_VOBU_SA_NOT_FOUND
The VMGM_VOBU_SA value could not be found in the VMGM_CP_SA list from the VMGM_C_ADT. Since every Cell starts with a VOBU, the VMGM_VOBU_SA should also be specified in the VMGM_CP_SA list.

9.3.15 DVD VTS checks

>>> [DVD] ERROR 4201 (ref. DVD-3 2.1) :
ERR_DVD_VTS_RESERVED_FIELD_ILL
All reserved fields should have all their bits cleared.

>>> [DVD] ERROR 4202 (ref. DVD-3 4.2) :
ERR_DVD_VTS_RESERVED_VALUE_ILL
All fields should not contain values which are reserved.

>>> [DVD] SYNTAX ERROR 4203 (ref. DVD-3 4.2.1) :
ERR_DVD_VTS_WRONG_OR_NO_VTSI
VTSI Parser Input: Probably a 'non-VTSI' stream!!!

>>> [DVD] ERROR 4210 (ref. DVD-3 4.2.1 / BP 0) :
ERR_DVD_VTS_ID_INVALID
The VTS_ID should describe "DVDVIDEO-VTS".

>>> [DVD] INFORMATION 4212 (ref. DVD-3 4.2.1 / BP 192) :
ERR_DVD_VTSM_VOBS_FOUND
Information message!
Reports that Menu-VOBS are found (VTSM_VOBS_SA > 0).

>>> [DVD] INFORMATION 4213: (ref. DVD-3 4.2.1 / BP 192) :
ERR_DVD_VTSM_VOBS_SA_ILL
VTSI_MAT: Allocation mismatch: VTSM_VOBS_SA has value <value>, it must be <value>, according the file start locations in UDF file system.

>>> [DVD] ERROR 4214 (ref. DVD-3 4.2.1 / BP 200) :
ERR_DVD_VTS_NO_VTSM_PGCI_UT
When Menu-VOBS exist, VTSM_PGCI_UT should exist as well.
(VTSM_PGCI_UT_SA > 0)

>>> [DVD] ERROR 4215 (ref. DVD-3 4.2.1 / BP 196) :
ERR_DVD_VTSTT_VOBS_SA_ILL
VTSI_MAT: Allocation mismatch: VTSTT_VOBS_SA has value <value>, it must be <value>, according the file start locations in UDF file system.

>>> [DVD] INFORMATION 4216 (ref. DVD-3 4.2.1 / BP 212) :
ERR_DVD_VTS_TMAPT_FOUND
Information message!
Reports that a Time Map Table was found in the VTSI (VTS_TMAPT_SA > 0).

>>> [DVD] ERROR 4218 (ref. DVD-3 4.2.1 / BP 216) :
ERR_DVD_VTS_NO_VTSM_C_ADT_SA
When Menu-VOBS exist, VTSM_C_ADT should exist as well.
(VTSM_C_ADT_SA > 0).

>>> [DVD] ERROR 4220 (ref. DVD-3 4.2.1 / BP 220) :
ERR_DVD_VTS_NO_VTSM_VOBU_ADMAP_SA
When Menu-VOBS exist, VTSM_VOBU_ADMAP should exist as well.
(VTSM_VOBU_ADMAP_SA > 0).

>>> [DVD] ERROR 4222 (ref. DVD-3 4.2.1 / BP 32) :
ERR_DVD_VTS_VERN_INVALID
The Version Number should be 1.1.

>>> [DVD] ERROR 4224 (ref. DVD-3 4.2.1) :
ERR_DVD_VTS_DATA_FOUND_WITHOUT_VOBS
When no Menu-VOBS exist, all Menu-related attributes should contain ‘0’.
(VTSM_V_ATR, VTSM_AST_ATR, VTSM_SPST_ATR.)

>>> [DVD] ERROR 4226 (ref. DVD-3 4.2.1 / BP 256 or 512) :
ERR_DVD_VTS_SOURCE_PIC_RES_RESERVED
The Source_picture_resolution field of VTS_V_ATR should only contain a specified value (0-3).

>>> [DVD] ERROR 4228 (ref. DVD-3 4.2.1 / BP 256 or 512) :
ERR_DVD_VTS_SOURCE_PIC_LETTBOX_ILL
Source_picture_letterboxed can only describe “Letterboxed” for Aspect_Ratio ‘0’ (4:3).

>>> [DVD] ERROR 4230 (ref. DVD-3 4.2.1 / BP 258) :
ERR_DVD_VTSM_AST_NS_TOO_LARGE
Only 0 or 1 Menu Audio streams can be specified.

>>> [DVD] ERROR 4232 (ref. DVD-3 4.2.1 / BP 258) :
ERR_DVD_VTSM_AST_NS_NOT_NULL (ref. DVD-3 4.2.1 / BP 258)
When no Menu-VOBS exist, the number of Audio Streams should be '0'.

>>> [DVD] ERROR 4234 (ref. DVD-3 4.2.1 / BP 260) :
ERR_DVD_VTSM_AUDIO_CODING_MODE_ILL
When TV_system decribes ‘0’ (ntsc), only ‘0’ (Dolby AC-3) or ‘1’ (Linear PCM) can be specified.
When TV_system decribes ‘1’ (pal), only ‘1’ (MPEG-1 or MPEG-2 without extension bitstream), ‘2’ (MPEG-2 with extension bitstream) or ‘3’ (Linear PCM) can be specified.

>>> [DVD] ERROR 4236 (ref. DVD-3 4.2.1 / BP 516) :

ERR_DVD_VTS_AUDIO_COD_MODE_ILL
When TV_system decribes ‘0’ (ntsc), only ‘0’ (Dolby AC-3) or ‘1’ (Linear PCM) can be specified.
When TV_system decribes ‘1’ (pal), only ‘1’ (MPEG-1 or MPEG-2 without extension bitstream), ‘2’ (MPEG-2 with extension bitstream) or ‘3’ (Linear PCM) can be specified.

>>> [DVD] ERROR 4238 (ref. DVD-3 4.2.1 / BP 260 or 516) :
ERR_DVD_VTS_DRC_RESERVED
The value of Dynamic_Range_Control in VTS(M)_AST_ATR is reserved for the specified Audio_coding_mode.

>>> [DVD] ERROR 4240 (ref. DVD-3 4.2.1 / BP 260 or 516) :
ERR_DVD_VTS_QUANTIZATION_RESERVED
The value of Quantization in VTS(M)_AST_ATR is reserved for the specified Audio_coding_mode.

>>> [DVD] ERROR 4242 (ref. DVD-3 4.2.1 / BP 260 or 516) :
ERR_DVD_VTS_FS_RESERVED
The value of fs in VTS(M)_AST_ATR can only be ‘0’ (48kHz) or ‘1’ (96 kHz).

>>> [DVD] ERROR 4244 (ref. DVD-3 4.2.1 / BP 260) :
ERR_DVD_VTS_AUDIO_CHANN_TOO_LARGE
The number of Audio Channels in VTSM_AST_ATR exceeds the maximum of 1 (2 channels) for the specified Audio_coding_mode.

>>> [DVD] ERROR 4246 (ref. DVD-3 4.2.1 / BP 340) :
ERR_DVD_VTSM_SPST_NS_TOO_LARGE
Only 1 Subpicture stream can be specified.

>>> [DVD] ERROR 4248 (ref. DVD-3 4.2.1 / BP 340) :
ERR_DVD_VTSM_SPST_NS_NOT_NULL
When no VTSM_VOBS exist, the number of Subpicture Streams should be '0'.

>>> [DVD] ERROR 4250 (ref. DVD-3 4.2.1 / BP 340) :
ERR_DVD_VTS_SPST_NS_ERR
When VTSM_VOBs exist, the number of Subpicture Streams should be '1'.

>>> [DVD] ERROR 4252 (ref. DVD-3 4.2.1 / BP 342) :
ERR_DVD_VTS_SUBPIC_CODING_MODE_RES
Sub_picture_coding_mode of VTSM_SPST_ATR should only specify ‘0’ or ‘1’.

>>> [DVD] ERROR 4254 (ref. DVD-3 4.2.1 / BP 512) :
ERR_DVD_VTS_FILM_CAMERA_MODE_ILL
Film_camera_mode can only specify ‘film mode’ for TV system 625/50 (pal).

>>> [DVD] ERROR 4256 (ref. DVD-3 4.2.1 / BP 514) :
ERR_DVD_VTS_AST_NS_TOO_LARGE
A maximum of 8 Audio streams may be specified.

>>> [DVD] ERROR 4258 (ref. DVD-3 4.2.1 / BP 514) :
ERR_DVD_VTS_AST_NS_NOT_NULL
When no VOBS exist, the number of Audio Streams should be '0'.

>>> [DVD] ERROR 4260 (ref. DVD-3 4.2.1) :
ERR_DVD_VTS_UNUSED_OBJ_CONTAINS_DATA
Unused fields should only contain ‘0’.
Only used in VTSI_MAT verification.

>>> [DVD] ERROR 4262 (ref. DVD-3 4.2.1 / BP 596) :
ERR_DVD_VTS_SPST_NS_TOO_LARGE
The maximum number of Subpicture streams shoud be 32.

>>> [DVD] ERROR 4266 (ref. DVD-3 4.2.2) :
ERR_DVD_VTS_TOO_MANY_PTT_IN_VTS
A total maximum of 999 PTTs can be in the TTUs.

>>> [DVD] ERROR 4268 (ref. DVD-3 4.2.2-1 / (2)) :

ERR_DVD_VTS_PTT_SRPT_EA_INVALID
The end of the PTT_SRPT is not found at the specified end address.

>>> [DVD] ERROR 4270 (ref. DVD-3 4.2.2-1 / (1)) :
ERR_DVD_VTS_TTU_NS_TOO_SMALL
The number of TTUs should equal the number of titles in this VTS.

>>> [DVD] ERROR 4272 (ref. DVD-3 4.2.2-1 / (1)) :
ERR_DVD_VTS_TTU_NS_TOO_LARGE
The maximum number of TTUs is 99.

>>> [DVD] ERROR 4274 (ref. DVD-3 4.2.6-1) :
ERR_DVD_VTS_NS_TOO_SMALL
The specified number is too small, should be at least 1.

>>> [DVD] ERROR 4275 (ref. DVD-3 4.2.2) :
ERR_DVD_VTS_PGN_ILLEGAL
PGN from TTU should be 1 when PGC is not a One_sequential_PGC_title.

>>> [DVD] ERROR 4276 (ref. VCPS G.2.2)
ERR_DVD_VTSI_VCPS_ID_INVALID
VTSI_MAT: The VCPS_ID is (<value>) and should be 'VCPS'.

>>> [DVD] ERROR 4280 (ref. DVD-3 4.2.3-1 / (2)) :
ERR_DVD_VTS_PGCIT_EA_INVALID
The specified end address of the VTS_PGCIT is incorrect.

>>> [DVD] ERROR 4282 (ref. DVD-3 4.2.3-1 / (1)) :
ERR_DVD_VTS_PGCI_SRP_Ns_TOO_SMALL
The number of PGCI Search pointers should be at least 1.

>>> [DVD] ERROR 4284 (ref. DVD-3 4.2.3-2 / (1)) :
ERR_DVD_VTS_PGC_CAT_VTS_TTN_TOO_SMALL
The title number is too small, should be at least 1.

>>> [DVD] ERROR 4286 (ref. DVD-3 4.2.3-2 / (1)) :
ERR_DVD_VTS_PGC_CAT_VTS_TTN_TOO_LARGE
The title number is too large, should be at most 99.

>>> [DVD] ERROR 4288 (ref. DVD-3 4.2.3-2 / (1)) :
ERR_DVD_VTSM_LCD_ILL
A language code should be between ‘AA’ and ‘ZZ’.

>>> [DVD] ERROR 4290 (ref. DVD-3 4.2.3-2 / (1)) :
ERR_DVD_VTSM_LCD_REUSED
A language code in VTSM_LCD is used more than once in this table.

>>> [DVD] ERROR 4292 (ref. DVD-3 4.2.6-2 or 4.2.8-2 / (3)) :
ERR_DVD_VTSM_CP_SA_IN_USE
The specified Start address was already used.

>>> [DVD] ERROR 4294 (ref. DVD-3 4.2.1)
ERR_DVD_VTS_TABLE_POS_ERR
This table was found at a different starting address than specified in the VTSI_MAT.

>>> [DVD] ERROR 4296 (ref. DVD-3 4.2.1)
ERR_DVD_VTS_RESERVED_BLOCK_ILL
A block of reserved bytes should only contain '0'.
Only used for reserved fields in VTSI_MAT.

>>> [DVD] ERROR 4298(ref. DVD-3 4.2)
ERR_DVD_VTS_EA_SMALL
The specified end address of the table is too small.

>>> [DVD] ERROR 4300 (ref. DVD-3 4.2)

ERR_DVD_VTS_EA_ERROR
The specified end address is not in accordance with the parsed length of the table.

>>> [DVD] ERROR 4302 (ref. DVD-3 4.2)
ERR_DVD_VTS_SA_ILLEGAL
The specified start address points to an illegal address.

>>> [DVD] ERROR 4304 (ref. DVD-3 4.2)
ERR_DVD_VTS_SA_ILL_ORDER
The specified start address is smaller than the previous start address.
In a table of start addresses, all addresses should be listed in ascending order.

>>> [DVD] ERROR 4306 (ref. DVD-3 4.2)
ERR_DVD_VTS_SRP_SA_INVALID
The specified start address does not correspond with the found start address.

>>> [DVD] ERROR 4308 (ref. DVD-3 4.2.4.1-2 / (1)) :
ERR_DVD_VTS_NO_ENTRY_MENU
No Entry Menu PGC was found in this table.

>>> [DVD] ERROR 4310 (ref. DVD-3 4.2.4.1-2 / (1)) :
ERR_DVD_VTS_MENU_NOT_FOUND
The VTSM_EXT specified an entry PGC exists for the menu_id, but this menu_id was not found.

>>> [DVD] ERROR 4312 (ref. DVD-3 4.2) :
ERR_DVD_VTS_SRP_ILL
The current start address is not found in the Search Pointer Table.

>>> [DVD] ERROR 4314 (ref. DVD-3 4.2.4.1-2 / (1)) :
ERR_DVD_VTS_MORE_ENTRY_MENU
More than one Entry Menu PGC was found in this table.

>>> [DVD] ERROR 4316 (ref. DVD-3 4.2.4.1-2 / (1)) :
ERR_DVD_VTS_MENU_ID_ILL
The specified Menu_ID is reserved for the corresponding Entry_type.

>>> [DVD] ERROR 4318 (ref. DVD-3 4.2.4.1-2 / (1)) :
ERR_DVD_VTS_BLOCK_MODE_ILL
For the corresponding Block_type, Block_mode should be the specified value.

>>> [DVD] ERROR 4320 (ref. DVD-3 4.2.4.1-2 / (1)) :
ERR_DVD_VTS_BLOCK_MODE_ILL2
The specified Block_mode should nt be used with the corresponding Block_type.

>>> [DVD] ERROR 4322 (ref. DVD-3 4.2.6-2 or 4.2.8-2 / (1)) :
ERR_DVD_VTS_VOB_IDN_ILL
VTS(M)_VOB_IDN is larger than the specified number of VOBs.

>>> [DVD] ERROR 4324 (ref. DVD-3 4.2.6-2 or 4.2.6-2 / (1)) :
ERR_DVD_VTS_VOB_IDN_ORDER
VTS(M)_VOB_IDN should be assigned continously.

>>> [DVD] ERROR 4326 (ref. DVD-3 4.2.6-2 or 4.2.8-2 / (2)) :
ERR_DVD_VTS_C_IDN_ORDER
All Cell ID numbers should be assigned continously.

>>> [DVD] ERROR 4328 (ref. DVD-3 4.2) :
ERR_DVD_VTS_EA_PAST_SA
The specified End Address points beyond the next Start Address.

>>> [DVD] ERROR 4330 (ref. DVD-3 4.2) :
ERR_DVD_VTS_EA_EQUAL_SA
The specified End Address cannot equal the next Start Address.

>>> [DVD] ERROR 4332 (ref. DVD-3 4.2.8-1) :

ERR_DVD_VTS_VOBU_NS_ILL
The specified number of VTSM_VOBs should be equal to, or larger than the number of VOBUs found in the VTSM_VOBU_ADMAP.

>>> [DVD] ERROR 4334 (ref. DVD-34.2.6-2 or 4.2.8-2 / (1)) :
ERR_DVD_VTS_VOB_IDN_DECREASE
VTS(M)_VOB_IDN cannot be assigned in decreasing order.

>>> [DVD] ERROR 4336 (ref. DVD-3 4.2.8-2) :
ERR_DVD_VTS_C_IDN_ILL
The specified Cell id should be equal to, or 1 higher than the previous Cell id.
Any other specified value is illegal.

>>> [DVD] ERROR 4338 (ref. DVD-34.2.8-2) :
ERR_DVD_VTS_CP_SA_LOWER_ILL
The specified start address cannot be lower than the previous start addres within a VOB.

>>> [DVD] ERROR 4340 (ref. DVD-3 4.2.8-2) :
ERR_DVD_VTS_CP_SA_EQUAL_ILL
The specified end address can only be equal to the previous end address, when a Cell boundary exists within a CellPiece.

>>> [DVD] ERROR 4342 (ref. DVD-3 4.2.8-2) :
ERR_DVD_VTS_CP_EA_ILL
The specified end address shall be equal to the previous end address, when a Cell boundary is detected in a CellPiece.

>>> [DVD] ERROR 4344 (ref. DVD-3 4.2.8-2) :
ERR_DVD_VTS_CP_EA_LOWER_SA
The specified end address should be larger than the start address.

>>> [DVD] ERROR 4346 (ref. DVD-3 4.2.7-1 or 4.2.9-1 / (1)) :
ERR_DVD_VTS_ADMAP_EA_OVER
The VOBU_ADMAP table exceeds the specified End Address.

>>> [DVD] ERROR 4348 (ref. DVD-3 4.2.7-1 / (1)) :
ERR_DVD_VTS_VOBU_SA_NOT_FOUND
The specified CellPiece Start Address was not found in the VOBU_ADMAP table.

>>> [DVD] ERROR 4350 (ref. DVD-3 4.2.7-1 / (1)) :
ERR_DVD_VTS_NEW_VOB_C_IDN_ILL
The first Cell ID number of a VOB should be '1'.

>>> [DVD] ERROR 4352 (ref. DVD-3 4.2.5) :
ERR_DVD_VTS_TMAP_TMU_INVALID
TMU should contain ‘0’ if no MAP_EN exists.

>>> [DVD] ERROR 4354 (ref. DVD-3 4.2.5) :
ERR_DVD_VTS_TMAP_MAP_EN_Ns_INVALID
The number of Map Entries should be ‘0’ when Time Unit is ‘0’.

>>> [DVD] ERROR 4356 (ref. DVD-3 4.2.5) :
ERR_DVD_VTS_TMAP_MAP_EN_Ns_TOO_LARGE (ref. DVD-3 4.2.5)
The number of Map Entries should be between 0 and 2048.

>>> [DVD] ERROR 4358 (ref. DVD-3 4.2) :
ERR_DVD_VTS_RESERVED_VALUE_ILL_STR (ref. DVD-3 4.2)
A field has a reserved value.
(This error is the same as ERR_DVD_VTS_RESERVED_VALUE_ILL, but the matching string of the reserved field is printed as well).

9.3.16 DVD PGCI checks

>>> [DVD] INFORMATION 4402 (ref. DVD-3 4.3.2) :

ERR_DVD_PGCI_DETECTED
This states that a PGC_CMDT, PGC_PGMAP, C_PBIT or C_POSIT was detected in the PGC_GI.

>>> [DVD] ERROR 4404 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_MANDATORY
A C_POSIT table is mandatory when a C_PBIT table exists.

>>> [DVD] INFORMATION 4406 (ref. DVD-3 4.3.2 (1)) :
ERR_DVD_PGCI_ILL_PRESENT
The presence of a C_POSIT is not allowed for the FP_PGCI.

>>> [DVD] ERROR 4408 (ref. DVD-3 4.3.2 (1)) :
ERR_DVD_PGCI_PGC_WO_VOB_DETECTED
The PGC in this PGC_CNT does not contain any VOB start addresses.

>>> [DVD] ERROR 4410 (ref. DVD-3 4.3.2 (1)) :
ERR_DVD_PGCI_NO_VOB
When no VOB is used in this PGC_CNT, the Number of Programs and Number of Cells should be '0'.

>>> [DVD] ERROR 4412 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_NOVOB_VALUE_EQ
If no VOB is used, the following fields should be ‘0’:

Number of Programs
Number of Cells
Hours (tens)
Hours (units)
Minutes (tens)
Minutes (units)
Second (tens)
Second (units)
Video frame (tens)
Video frame (units)

>>> [DVD] ERROR 4414 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_RANGE
This message is used to state any errors in the valid range of:

Number of Programs	0 .. 99
Number of Cells	0 .. 255
Hours (tens)	0 .. 9
Hours (units)	0 .. 9
Minutes (tens)	0 .. 5
Minutes (units)	0 .. 9
Second (tens)	0 .. 5
Second (units)	0 .. 9
Video frame (tens)	0 .. 2
Video frame (units)	0 .. 9

>>> [DVD] ERROR 4416 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_RESERVED
This field is using a reserved value, which is not allowed.

>>> [DVD] ERROR 4418 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_MAX_NUM
The maximum number of PRE_CMD_Ns is 128.

The maximum number of POST_CMD_Ns is 128.

The maximum number of C_CMD_Ns is 128.

>>> [DVD] ERROR 4420 (ref. DVD-3 4.3.2) :

ERR_DVD_PGCI_MAX_SUM
The sum of PRE_CMD_Ns + POST_CMD_Ns + C_CMD_Ns should be at most 128.

>>> [DVD] ERROR 4422 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_TOO_LARGE
The specified PGC_CMDT_EA is beyond the actual PGC_CMD table size.

>>> [DVD] ERROR 4424 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_TOO_SMALL
The specified PGC_CMDT_EA is smaller than the actual PGC_CMD table size.

>>> [DVD] ERROR 4426 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_ALWAYS_VALUE
The first EN_CN of the PGC_PGMAP should always be '1'.

>>> [DVD] ERROR 4428 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_DEPEND_VALUE
If C_PBIT_SA is '0', Luminance_signal_Y, Color_difference_Cr and Color_difference_Cb should be '0' as well.

>>> [DVD] ERROR 4430 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_CELL_BLOCK_EQ (ref. DVD-3 4.3.2)
If Cell_Block_type is '1' (Angle block), Cell_Block_mode should not be '0' (Not a cell in the block).

>>> [DVD] ERROR 4432 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_CELL_BLOCK_NEQ (ref. DVD-3 4.3.2)
If Cell_Block_type is '0', Cell_Block_mode should not be '0'.

>>> [DVD] ERROR 4434 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_RES_FIELD_NOT_ZERO
All Reserved fields should contain only '0'.

>>> [DVD] ERROR 4436 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_CELL_TYPE_ERR
The Cell_type should contain '0' when Application type is NOT Karaoke.

>>> [DVD] ERROR 4438 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_DEC_STR_ERR
Decoding Audio or Sub_picture stream numbers should be '0' when Availability flag is '0'.

>>> [DVD] ERROR 4440 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_COMMAND_NR_TOO_LARGE
The Cell_Command_Number is larger than number of Cell Commands.

>>> [DVD] ERROR 4442 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_CELL_PB_MODE_ERR
The Cell_playback_mode should be set to '0' when Cell_Still_time is not set to '0'.

>>> [DVD] ERROR 4444 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_LAST_CELL_ERR
When Still time value is NOT set to '0' the last Cell in all PGs should be set to '0'.

>>> [DVD] ERROR 4446 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_DOMAIN_ERR
This errormessage is used to state that:

- The Next_PGC_number should be '0' when found in a PGC in the System space.
- The Previous_PGC_number should be '0' when found in a PGC in the System space.
- The GoUp_PGCN should be '0' when found in a PGC in the FP_DOM space.
- The PG_Playback_mode should be '0' when found in a PGC in the Menu space.

>>> [DVD] ERROR 4447 (ref. DVD-3 4.3.2) :

ERR_DVD_PGCI_DATA_NO_VOB
The Availability flag for 'audio or sub-picture' should be '0' when found in a PGC in without any Cells or Programs.

>>> [DVD] ERROR 4448 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_PGC_PB_TM_ILL
PGC_GI.PGC_PB_TM (hh:mm:ss.ff) must be equal to the sum the C_PBTM of all the Cells belonging to this PGC.
Mismatch between PGCI PBTM and sum of C_PBTM's of cells belonging to the same PGC (Title).

>>> [DVD] ERROR 4449 (ref. DVD-3 4.3.2) :
ERR_DVD_PGCI_CELL_SEAMLESS_PLBCK
When PGC playback mode is Random (0x01 ...0x07h) or Shuffle (0x81...0xFFh), the seamless playback flag in first cell of every Program must be set to '0'.

9.3.17 DVD PCI checks

Assumptions

[A1] The video field grid starts on the first PTS (or DTS) of the video data. Since the first VOBU has to contain the first video, this is the first PTS of the first VOBU.

[A2] HLI information is only allowed in a menu VOBS (in Menu Space).

[A3] A VOBU's video presentation start time is given by the presentation start time of its first picture in DISPLAY ORDER ! Notice that in coding order this first picture (which is always an I-picture) may be preceded by some B-pictures.

[A4] Angle changes are confined to the current VOB.

>>> [DVD] INFORMATION 4501 (ref. N/A) :
ERR_DVD_PCI_NO_XCHECK_PARAS
Necessary cross check parameters not found on the cross check data file ! Certain PCI checks will use default values for missing cross check parameters:

cross check parameter	default value
Cell_still_time	0
Cell_Block_type	0
Seamless_Angle_Change_flag	1
Number of Angles	1

9.3.17.1 PCI_GI Checks

>>> [DVD] ERROR 4511 (ref. DVD-3 4.4.1 (1)) :
ERR_DVD_PCI_GI_NV_PCK_LBN
The NV_PCK_LBN value is not equal to the RLBN of the NV_PCK this PCI is included in.

>>> [DVD] ERROR 4512 (ref. DVD-3 4.4.1 (2)) :
ERR_DVD_PCI_VOBU_CAT
The APSTB value should be 0 when the CGMS in the file descriptor of the file containing this VOBU, is not 0x11.

This is a Cross Check between the File System & VOB data !

Not implemented yet : CGMS of file descriptor not available yet.

>>> [DVD] ERROR 4513 (ref. DVD-3 4.4.1 (3) / Table J.2-1) :

ERR_DVD_PCI_VOBU_UOP_CTL
The indicated VOBU_UOP_CTL_UOP bit is 1, should be 0 (reserved).

>>> [DVD] ERROR 4521 (ref. DVD-3 4.4.1 (4)) :
ERR_DVD_PCI_VOBU_S_PTM
The VOBU contains video and the VOBU_S_PTM value is different from the VOBU's video presentation start time.

Taking into account [A3], the latter is given by :

- PTS of the first picture when it has a temporal reference zero, since this indicates it is the 1st picture in display order.
- DTS of the first picture to which 1 frame period is added.

>>> [DVD] ERROR 4522 (ref. DVD-3 4.4.1 (4)) :
ERR_DVD_PCI_VOBU_S_PTM_MULT
The VOBU does not contain video data and the VOBU_S_PTM value is not aligned with the video field grid. (Assuming [A1]).

>>> [DVD] ERROR 4523 (ref. DVD-3 ???)) :
ERR_DVD_PCI_VOBU_S_PTM_ILL
Illegal VOBU_S_PTM value : smaller than the previous VOBU's NV_PCK SCR.

- It is not yet clear how the VOBU's start & termination times are constrained by the pack's SCR values.

>>> [DVD] ERROR 4524 (ref. DVD-3 4.4.1 (5)) :
ERR_DVD_PCI_VOBU_E_PTM
The VOBU_E_PTM value is different from the VOBU 's video presentation termination time.

>>> [DVD] ERROR 4525 (ref. DVD-3 4.4.1 (6)) :
ERR_DVD_PCI_VOBU_E_PTM_MULT
The VOBU does not contain video data or is terminated (with a sequence_end_code) and the VOBU_E_PTM value is not aligned with the video field grid. (Assuming [A1]).

>>> [DVD] ERROR 4526 (ref. DVD-3 ???) :
ERR_DVD_PCI_VOBU_E_PTM_ILL
Illegal VOBU_E_PTM value : larger than the next VOBU 's NV_PCK.

>>> [DVD] ERROR 4527 (ref. DVD-3 4.4.1 (6)) :
ERR_DVD_PCI_VOBU_SE_E_PTM
The VOBU_SE_E_PTM value is different from the VOBU's video presentation termination time.

>>> [DVD] ERROR 4528 (ref. DVD-3 4.4.1 (6)) :
ERR_DVD_PCI_VOBU_SE_E_PTM_0
The VOBU_SE_E_PTM value is not zero while the VOBU does not contain a sequence_end_code.

>>> [DVD] ODDITY 4529 (ref. DVD-3 4.4.1 (4,5)) :
ERR_DVD_PCI_VOBU_PTM_DUR
PCI_GI : The VOBU_S_PTM value 'value' and VOBU_E_PTM value 'value' specify a VOBU presentation time 'value' which is no integer multiple of the video frame period 'value'.

>>> [DVD] ERROR 4531 (ref. DVD-3 4.4.1 (7)) :
ERR_DVD_PCI_GI_C_ELTM_ILL
One of the C_ELTM BCD field contains an illegal value, i.e. not within the specified (hour, minute, second) boundaries.

>>> [DVD] ERROR 4532 (ref. DVD-3 4.4.1 (7)) :
ERR_DVD_PCI_GI_C_ELTM_RSRVD
C_ELTM contains a reserved tc_flag value 0x00 or 0x10.

>>> [DVD] ERROR 4533 (ref. DVD-3 4.4.1 (7)) :

ERR_DVD_PCI_GI_C_ELTM_TVSYS

C_ELTM contains a tc_flag value which is inconsistent with the stream's TV system (PAL or NTSC).

>>> [DVD] ERROR 4534 (ref. DVD-3 4.4.1 (7)) :
ERR_DVD_PCI_GI_C_ELTM_1ST

C_ELTM value is not zero for the first PCI of a Cell.

>>> [DVD] ERROR 4535 (ref. DVD-3 4.4.1 (7)) :
ERR_DVD_PCI_GI_C_ELTM

C_ELTM value has a different value than expected. The expected value is the duration of the number of (real or imagined) video frames between the start of the Cell and the current VOBU.

>>> [DVD] ERROR 4538 (ref. DVD-3 4.4.1 (7)) :
ERR_DVD_PCI_GI_C_ELTM_XCHK

The C_ELTM tc_flag value is different from that specified by occurrences in the navigation data, i.e. PGCI data :

- PGC_PB_TM (cf. [DVD-3] 4.3.2 (2))
- C_PBTM (cf. [DVD-3] 4.3.5 (2))

This is a Cross Check between the disk's Navigation & VOB data !

Not implemented yet.

9.3.17.2 NSML_AGLI Checks

Observations :

1. An Angle Block (AGL_C_BLK) consists of max. 9 Angles, each composed of exactly 1 Angle Cell (AGL_C).
2. One AGL_C consists of an integer number of "parts of AGL_C", of which the start address is described by the PCI NSML_AGLI data.
3. The NSML_AGLI data describes a sequential 'slice' of a complete Angle Block, always containing a part (of identical duration) of each of the 9 possible Angle Cells.
4. Each (part of) AGL_C consists of an integer number of VOBUs, possibly more than 1.
5. Angle Cells of a non-Seamless Angle Block are multiplexed as complete & consecutive data : i.e. all VOBUs part of the same Angle Cell follow each other; after the last VOBU of an Angle Cell, the first VOBU of the next Angle Cell starts.
6. Each Angle Cell is in fact another VOB within a non-Seamless Angle Block.

Used Cross Check Parameters :

A dedicated flag "Seamless_Angle_Change_flag" & parameter "Cell_Block_type" have been defined & used for these checks. These match a field with the same name of the PGCI - C_PBI - C_CAT data structure (cf. [DVD-3] Table 4.3.5-1 (1)) and is made available through the Xcheck data file. Also the number of Angles defined in the current Title (defined by the VMGI - TT_SRPT - TT_SRP(i) - AGL_Ns field, cf.[DVD-3] Table 4.1.2-2) is passed through the Xcheck data file.

As a consequence, the checks using any of these parameters can only be properly performed when the proper Cross Checks data (file) is present. If this file is missing, rather than disabling these checks, the necessary parameters otherwise retrieved from this file, are given their default value (which in most cases comes down to a de facto disabling of the checks) :

Seamless_Angle_Change_flag	1
Cell_Block_type	0
AGL_Ns	1

PCI NSML_AGLI Verification Lists :

The verification of the NSML_AGLI table entries is done separately for forward & backward references :

- Backward references (i.e. references to VOBUs containing or before the current PCI & marked by AGL_C location == 1) are checked immediately using a "VOBU" list with all the VOBUs encountered in the current VOB so far. Each entry contains all necessary information for the checks (absolute pack (LB) address, start time, etc.).
- Forward references (i.e. references to VOBUs after the current PCI's stream position & marked by AGL_C location == 0) are checked whenever the target VOBU, i.e. the VOBU referenced by a PCI in the preceding part of the AGL_C_BLK, is encountered in the stream. Therefore a "ref" list is used to store all forward NSML_AGLI references still to be checked.

  Since all these references are restricted to the current Angle Block, this "ref" list is generated during the current AGL_C_BLK parsing, valid for the current block only and destroyed at the end of the AGL_C_BLK.

>>> [DVD] ERROR 4541 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_NOT0 (ref. DVD-3 4.4.2)
Not all 9 NSML_AGLI_C[#n]_DSTA entries are zero, although no Angle Block exists or the Angle Block is seamless.

>>> [DVD] ERROR 4542 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_NR
The n-th NSML_AGLI_C[#n]_DSTA entry contains a non-zero value, although there are fewer than n angles defined (by the VMGI - TT_SRPT - TT_SRP(i) - AGL_Ns field, cf. [DVD-3] Table 4.1.2-2).

>>> [DVD] ERROR 4543 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_0
The n-th NSML_AGLI_C[#n]_DSTA entry contains a zero value, although a Non-Seamless Angle Change has been indicated defined (by the VMGI - TT_SRPT - TT_SRP(i) - AGL_Ns field, cf. [DVD-3] Table 4.1.2-2).

>>> [DVD] ERROR 4544 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_LOC
The n-th NSML_AGLI_C[#n]_DSTA entry AGL_C location field indicates a VOBU location after/before the current NV_PCK, while it is before/after.

>>> [DVD] ERROR 4545 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_STRT
The n-th NSML_AGLI_C[#n]_DSTA entry AGL_C field specifies an incorrect (non-existing) VOBU start address.

>>> [DVD] ERROR 4546 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_PST
The presentation start time of the VOBU described by the n-th NSML_AGLI_C#n_DSTA ACL_C, should be equal or immediately before/after the presentation start time of the current VOBU (containing this PCI).

>>> [DVD] ERROR 4547 (ref. DVD-3 4.4.2) :
ERR_DVD_PCI_NSML_AGLI_NO_PST
The PCI of the current VOBU of an Angle Cell, with start PTM ≤ 1.2 seconds less than the Cell end PTM, does not have its AGL_C destination address field set to 0x7FFFFFFF to prevent angle changes at the end of an Angle Block.

This is no ERROR, but implemented as an RECOMMENDATION VIOLATION.

- This check is triggered by an EVT_VOB_END event. Taking into account observation-6, this marks the end of an AGL_C. All (still unverified) references still stored in the reference list are then considered to belong to the last AGL_C part and to refer to no other AGL_C part. These should then have their AGL_C destination address field set to 0x7FFFFFFF.

>>> [DVD] ERROR 4548 (ref. DVD-3 4.4.2) :

ERR_DVD_PCI_NSML_AGLI_AGL_C_NOT1

The n-th NSML_AGLI_C[#n]_DSTA entry AGL_C destination address field is not 0x7FFFFFFF for the current PCI following a VOBU with start PTM ≤ 1.2 seconds less than the Cell end PTM which has its AGL_C destination address field set to 0x7FFFFFFF.

This check together with the previous one originate from the v1.0 spec update and are interpreted as follows : **all** VOBUs (PCIs) of the **last** Angle Cell piece of **each** Angle Cell of a non-Seamless Angle Block, should have their AGL_C destination address field set to 0x7FFFFFFF or **none**.

9.3.17.3 HL_GI Checks

Observations :

1. The presentation start time of a SPU is the PTS of the packet containing its first byte.

Assuming [A2] : VMGI_MAT - VMGM_SPST_Ns & VTSI_MAT - VTSM_SPST_Ns allow max. 1 SP stream. So the HLI info matches the only SP present and there is no need to specify which SP this HLI data is aimed at. However :

One SP stream may contain several sub-picture units (SPUs), when e.g. for proper multiplexing SP data is transmitted well before it is needed. In this case, the HLI is aimed at the SPU for which the presentation time is within the VOBU 's presentation time. So the SPU matches the VOBU (and its matching HLI) for which :

$$VOBU\_S\_PTM \le SPU\_PTS \le VOBU\_E\_PTM$$

2. When a SPU has a STP_DSP command, this has to be its last SP_DCSQ command (cf. [DVD-3] Annex L, Table L-1 5.d). As a consequence, the "start time of the DCSQ (SP_DCSQ_STM)" equals the SPU presentation termination time SPU_PTT.

   Where SPU_PTT = SPU_PST + SP_DCSQ_STM*1024.

3. cf. [DVD-3] Table 5.1-1 : A Sub-picture presentation is valid only in the Cell where the SPU is recorded. This means a sub-picture can not pass Cell boundaries and VOBU - SPU matches are confined to the same Cell.

Cross Check Parameters :

Again a dedicated parameter "Cell_Still_time" has been defined & used for these checks. It matches a field with the same name of the PGCI - C_PBI - C_CAT data structure (cf. [DVD-3] Table 4.3.5-1 (1)) and is made available through the Xcheck data file.

As a consequence, the checks using these parameters can only be properly done when the proper Cross Checks data (file) is present. If this file is missing, rather than disabling these checks, the necessary parameter otherwise retrieved from this file, is given its default value (which in most cases comes down to a de facto disabling of the checks) :

Cell_Still_time	0

Verification Lists :

Two lists are used for verification of PCI HL_GI data and matching SPUs :

1. The "SPUL" list stores relevant data of SPUs, multiplexed in the stream well before they are needed, thus ahead of the VOBU PCI containing the matching HLI data.
2. The "PCIL" list, storing the relevant VOBU and PCI HLI data for PCIs with HLI data still unmatched by the underlying SPU.

Both lists are used at the appropriate moments (events) to verify the correct match of HLI and SPU data.

>>> [DVD] ERROR 4550 (ref. DVD-3 4.4.3.2 (1)) :

ERR_DVD_PCI_HLI_1ST

HLI_SS is not 0x0 or 0x1 as is required for the first VOBU of a Cell and a fortiori for the first VOBU of a VOB.

>>> [DVD] ERROR 4551 (ref. DVD-3 4.4.3.2 (1)) :
ERR_DVD_PCI_HLI_SS_COPY

HLI_SS (01b) indicates that the HLI data should be different from that of the previous VOBU within the current Cell / VOB, but it is identical !

- This includes the following HLI data :
  - the HL_GI data
  - the BTN_COLIT data
  - the BTNIT data

>>> [DVD] ERROR 4552 (ref. DVD-3 4.4.3.2 (1)) :
ERR_DVD_PCI_HLI_SS_DIFF

HLI_SS (10b or 11b) indicates that the HLI data should be identical to that of the previous VOBU, which is not the case.

- This includes the following HLI data :
  - the HL_GI data (except for the HLI_SS value)
  - the BTN_COLIT data
  - the BTNIT data, except for the BTN_CMD data when HLI_SS is 11b

>>> [DVD] ERROR 4553 (ref. DVD-3 4.4.3.2 (1) & 3.3.10.1 2nd Note) :
ERR_DVD_PCI_HLI_DIFF_ SML

In a Seamless Angle Block completely identical HLI data has to be recorded.

! Not implemented yet.

>>> [DVD] ERROR 4556 (ref. DVD-3 4.4.3.2 (2) & 3.3.10.1) :
ERR_DVD_PCI_HLI_S_PTM

The HLI_S_PTM value is not equal to the presentation start time of the SPU it is aimed at (SPU_PST).

- As a consequence of Observation-1 above, at most the first SPU can be verified immediately against the PCI HLI data of the VOBU it is contained in (i.e. upon an EVT_SPU_START event). Other SPU data is to be stored and verified when its matching HLI data becomes available (upon a new EVT_PCI event).

Where the presentation start time of a SPU is the PTS of the packet containing its first byte.

>>> [DVD] ERROR 4557 (ref. DVD-3 4.4.3.2 (3) & 3.3.10.1) :
ERR_DVD_PCI_HLI_E_PTM

The HLI_E_PTM value is not equal to the presentation termination time of the SPU it is aimed at.

This error message can be generated in three distinct situations (cf. [DVD-3] 3.3.10.1) :

1. If the SPU is not the last of a Cell and it has no STP_DSP command, when the HLI_E_PTM value is different from the PTS of the next SPU.
2. If the SPU has a STP_DSP command, when the HLI_E_PTM value is different from the SPU's presentation termination time.
3. If the corresponding SPU is the last in the Cell and it has no STP_DSP command, when the HLI_E_PTM value is different from the end time of the Cell's last VOBU (PCI_GI - VOBU_E_PTM).

- This is verified using the above described verification lists, to match PCI HLI data and SPU data. Again this has to be verified on distinct moments :
  - for current or future SPUs, using the PCIL list
    - on EVT_SPU_START (situation 1)

    - on EVT_SPU (situation 2)
- for “send-ahead” SPUs, using the SPUL list
    - on EVT_PCI (situation 1 or 2)
- on EVT_CELL_END (for situation 3)

>>> [DVD] ERROR 4558 (ref. DVD-3 4.4.3.2 (4) & 3.3.10.1) :
ERR_DVD_PCI_HLI_SL_E_PTM
This error message is generated in two distinct situations :

1. The HLI_SL_E_PTM value is not larger than the SPU highlight start time (HLI_S_PTM).
2. The HLI_SL_E_PTM value is not smaller than or equal to the SPU highlight termination time (HLI_E_PTM).

>>> [DVD] ERROR 4559 (ref. DVD-3 4.4.4.2 (3,4) & 3.3.10.1) :
ERR_DVD_PCI_HLI_E_PTM_STILL
The HLI_E_PTM or BTN_SL_E_PTM value is not 0xFFFFFFFF during a Cell Still, or HLI_E_PTM equals 0xFFFFFFFF and BTN_SL_E_PTM does not.

>>> [DVD] ERROR 4560 (ref. DVD-3 4.4.3.1) :
ERR_DVD_PCI_HLI_BTNGR_CONT_DIF
Corresponding buttons in distinct button groups, i.e. buttons sharing the same button number, differ more than for their display position or size.

>>> [DVD] ERROR 4561 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_BTNGR_NS_0
The number of button groups BTNGR_Ns should be larger than 0, for valid HLI data.

>>> [DVD] ERROR 4562 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_BTNGR_NS_43
When the (VTS_V_ATR) Video attribute aspect ratio is 4:3, there can be only 1 button group : BTNGR_Ns should be 01b.

>>> [DVD] ERROR 4563 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_BTNGR1_DSP_43 (ref. DVD-3 4.4.3.2 (5))
When the (VTS_V_ATR) Video attribute aspect ratio is 4:3, the only button group present should have a zero display type of the Decoding SP stream (meaning only Normal 4:3 presentation).

>>> [DVD] ERROR 4564 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_BTNGR1_DSP_16_9 (ref. DVD-3 4.4.3.2 (5))
When the (VTS_V_ATR) Video attribute aspect ratio is 16:9, only Normal 4:3 presentation display type of the Decoding SP stream is not allowed, even for group 1.

>>> [DVD] ERROR 4565 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_BTNGR23_DSP (ref. DVD-3 4.4.3.2 (5))
The button group display type for non-existing button groups should be 0 :

BTNGR[#n]_DSP_TY should be 0x0 when BTNGR_Ns < i.

>>> [DVD] ERROR 4566 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_BTNGR23_DSP0 (ref. DVD-3 4.4.3.2 (5))
The button group display type for button groups 2 or 3 should not be 0 when these groups have been defined :

BTNGR2_DSP_TY should be > 0 when BTNGR_Ns >= 2, and
BTNGR3_DSP_TY should be > 0 when BTNGR_Ns = 3.

>>> [DVD] ERROR 4567 (ref. DVD-3 4.4.3.2 (5) Note) :
ERR_DVD_PCI_HLI_BTNGR_DSP_DUP
The same Decoding SP stream display type should not be used by different button groups.

>>> [DVD] ERROR 4568 (ref. DVD-3 4.4.3.2 (7..10)) :
ERR_DVD_PCI_HLI_BTN_NS
The specified number of buttons is larger than allowed :

- For the number of valid buttons (BTN_Ns), at least 1 and max 36, 18 or 12 for resp. 1, 2 or 3 button groups.
- For the U_BTNN selectable number of buttons (NSL_BTN_Ns), max 36, 18 or 12 for resp. 1, 2 or 3 button groups.
- For forcedly selected number of buttons (FOSL_BTNN), max 36, 18 or 12 for resp. 1, 2 or 3 button groups.
- For the forcedly activated number of buttons (FOAC_BTNN), max 36, 18 or 12 for resp. 1, 2 or 3 button groups or 63.

>>> [DVD] ERROR 4569 (ref. DVD-3 4.4.3.2 (9)) :
ERR_DVD_PCI_HLI_NSL_BTN_NS
The U_BTNN selectable number of buttons (NSL_BTN_Ns) is larger than the number of buttons per button group (BTN_Ns).

>>> [DVD] ERROR 4570 (ref. DVD-3 4.4.3.2 (5)) :
ERR_DVD_PCI_HLI_NO_BTNGR_DSP
The display mode of the (VTS_V_ATR) Video attribute allows for Pan-scan or Letterbox presentation and no matching button group is defined.

9.3.17.4 BTNIT Checks

>>> [DVD] ERROR 4571 (ref. DVD-3 4.4.3.4 (a)) :
ERR_DVD_PCI_HLI_BTN_COLN
The button's Button Color number should be 1,2 or 3.

>>> [DVD] ERROR 4572 (ref. DVD-3 4.4.3.4 (a)) :
ERR_DVD_PCI_HLI_BTN_POSI_ACT
A button's BTN_POSI specifies a reserved Auto action mode value (> 01b).

>>> [DVD] ERROR 4573 (ref. DVD-3 4.4.3.4 (a)) :
ERR_DVD_PCI_HLI_BTN_POSI
A button's BTN_POSI specifies an X or Y, start or end-coordinate outside the valid range :

0..719 for X-coordinates
2..479 or 2..574 for Y-coordinates in resp. 525/60 or 625/50 TV systems.

>>> [DVD] ERROR 4576 (ref. DVD-3 4.4.3.4 (b)) :
ERR_DVD_PCI_HLI_AJBTN_POSI_ILL
A button's AJBTN_POSI specifies a button number outside the valid range for the number of groups defined (i.e. 1..12,18 or 36 for resp. 1,2 or 3 groups).

>>> [DVD] ERROR 4577 (ref. DVD-3 4.4.3.4 (b)) :
ERR_DVD_PCI_HLI_AJBTN_POSI_DEF
A button's AJBTN_POSI specifies an undefined Button number, i.e. a button whose BTNI fields are all

9.3.17.5 RECI Checks

>>> [DVD] ERROR 4581 (ref. DVD-3 Table T-1) :
ERR_DVD_PCI_ISRC_CHAR_CODE
One of the ISRC fields for Country Code or Copyright Holder Code, specifies a character code other than these specified by Annex T, Table T-1 (should be 0..9 or 17..42).

>>> [DVD] ERROR 4582 (ref. DVD-3 Annex T Note 2) :
ERR_DVD_PCI_ISRC_BCD_CODE
One of the ISRC fields for Recording Year or Recording Number, specifies a non-BCD code.

>>> [DVD] ERROR 4585 (ref. DVD-3 4.4.4 (4)) :
ERR_DVD_PCI_ISRC_SP_SEL_GR_SET
More than one SP_GR[#n] group is set to 1.

9.3.18 DVD DSI checks

Assumptions

[A1] Angle change jumps are restricted to the current VOB.

>>> [DVD] INFORMATION 4601 (ref. N/A) :
ERR_DVD_DSI_NO_XCHECK_PARAS
Necessary cross check parameters not found on the cross check data file ! Certain DSI checks will use default values for missing cross check parameters:

cross check parameter	default value
Cell_Block_type	0
Number of Audio streams	dvd->VTS_AST_Ns
Number of Sub-picture streams	0
Seamless_Angle_Change_flag	0
Number of Angles	1

9.3.18.1 DSI_GI Checks

>>> [DVD] ERROR 4610 (ref. DVD-3 4.5.1 (1)) :
ERR_DVD_DSI_NV_PCK_SCR
The NV_PCK_SCR value is different from the SCR lower 32 bit SCR_base value of the NV_PCK which contains this DSI.

>>> [DVD] ERROR 4611 (ref. DVD-3 4.5.1 (2)) :
ERR_DVD_DSI_NV_PCK_LBN
The NV_PCK_LBN value is different from the relative address of the VOBU which contains this DSI.

>>> [DVD] ERROR 4612 (ref. DVD-3 4.5.1 (3)) :
ERR_DVD_DSI_VOBU_EA
The VOBU_EA value is different from the relative address of the last pack of the VOBU which contains this DSI.

- Here "relative address" is the number of packs from the start of the VOBU containing this DSI; i.o.w. it is the number of packs in the current VOBU, minus one.
- Checked at the end of a VOBU.

>>> [DVD] ERROR 4614 (ref. DVD-3 4.5.1 (4,5,6)) :

ERR_DVD_DSI_VOBU_REF_EA
This message covers incorrect addresses in 3 distinct data fields :

1. VOBU_1STREF_EA is not the relative address of the video pack containing the last data byte of the first encoded reference picture, which is an I-picture.
2. VOBU_2NDREF_EA is not the relative address of the video pack containing the last data byte of the 2nd encoded reference picture, which might be an I- or P-picture.
3. VOBU_3RDREF_EA is not the relative address of the video pack containing the last data byte of the 3rd encoded reference picture, which might be an I- or P-picture.

- Here "relative address" is the number of packs from the start of the VOBU containing this DSI; i.o.w. it is the number of packs between the start of the current VOBU and the specified video pack, minus one.
- These addresses do not have to be different, e.g. in case of very 'small' pictures.
- Checked at the end of a VOBU.

>>> [DVD] ERROR 4615 (ref. DVD-3 4.5.1 (4,5,6)) :
ERR_DVD_DSI_VOBU_REF_ORD
This message covers 2 distinct errors :

1. The VOBU_2NDREF_EA value is smaller than the VOBU_1STREF_EA value, or VOBU_3RDREF_EA smaller than VOBU_1STREF_EA or VOBU_2NDREF_EA.
2. One of the data fields VOBU_1STREF_EA, VOBU_2NDREF_EA or VOBU_3RDREF_EA is larger than the VOBU_EA value.

>>> [DVD] ERROR 4616 (ref. DVD-3 4.5.1 (4,5,6)) :
ERR_DVD_DSI_VOBU_REF_0
This message covers errors in 3 distinct data fields :

1. VOBU_1STREF_EA is not zero and the VOBU contains no video data (thus no I-picture).
2. VOBU_2NDREF_EA is not zero and the VOBU has no 2nd reference picture encoded.
3. VOBU_3RDREF_EA is not zero and the VOBU has no 3rd reference picture encoded.

- Checked at the end of a VOBU.

>>> [DVD] ERROR 4617 (ref. DVD-3 4.5.1 (7)) :
ERR_DVD_DSI_VOBU_VOB_IDN
An illegal VOB ID number VOBU_VOB_IDN has been specified. This message covers two possible errors :

1. Zero ID number
2. ID number which is more than 1 higher than the previous value.

>>> [DVD] ERROR 4618 (ref. DVD-3 4.5.1 (8)) :
ERR_DVD_DSI_VOBU_C_IDN
An illegal Cell ID number VOBU_C_IDN has been specified. This message covers two possible errors :

1. Zero ID number
2. ID number which is more than 1 higher than the previous value.

>>> [DVD] ERROR 4619 (ref. DVD-3 4.5.1 (9)) :
ERR_DVD_DSI_VOBU_C_ELTM
The specified C_ELTM value in this DSI is different from the C_ELTM value specified in the PCI.

9.3.18.2 SML_PBI Checks

>>> [DVD] ERROR 4621 (ref. DVD-3 4.5.2 (1)) :

ERR_DVD_DSI_VOB_NO_PREU
A VOB is allocated in a Contiguous Block and connected seamlessly with the next VOB in an Interleaved Block, and the former VOB is not defined as PREU.

>>> [DVD] ERROR 4622 (ref. DVD-3 4.5.2 (1)) :
ERR_DVD_DSI_PREU_SHORT
The PREU contains not enough VOBUs to cover at least 75 or 90 video display fields for resp. a 625/50 (PAL) or 525/60 (NTSC) TV system.

>>> [DVD] ERROR 4623 (ref. DVD-3 4.5.2) :
ERR_DVD_DSI_PREU_DUR_SHORT
Combined duration of the VOBUs in the PREU is less then the time necessary to contain the necessary display fields.

>>> [DVD] ERROR 4624 (ref. DVD-3 4.5.2 (1)) :
ERR_DVD_DSI_PREU_ILVU_FLAG
This error message is generated in 2 distinct cases :

1. The PREU flag is set, but this VOBU is not part of a PREU, or the other way around.
2. The ILVU flag is set, but this VOBU is not part of a ILVU, or the other way around.

>>> [DVD] ERROR 4625 (ref. DVD-3 4.5.2 (1)) :
ERR_DVD_DSI_PREU_OR_ILVU
PREU and ILVU flags have both been set, which is not allowed.

>>> [DVD] ERROR 4626 (ref. DVD-3 4.5.2 (1)) :
ERR_DVD_DSI_UNIT_STRT_END
This error message is generated in 4 distinct cases :

1. The Unit Start flag is set, but this VOBU is not at the beginning of a PREU or ILVU.
2. The Unit End flag is set, but this VOBU is not at the end of a PREU or ILVU.
3. The Unit Start flag is not set, but this VOBU is at the beginning of a PREU or an ILVU.
4. The Unit End flag is not set, but this VOBU is at the end of a PREU or an ILVU.

>>> [DVD] ERROR 4627 (ref. DVD-3 4.5.2 (1)) :
ERR_DVD_DSI_UNIT_STRT_OR_END
Unit Start and End flags have both been set, which is not allowed.

>>> [DVD] ERROR 4629 (ref. DVD-3 4.5.2 (1)) :
ERR_DVD_DSI_PREU_SEQEND
A sequence_end_code occurs in this VOB which has been defined as PREU.

>>> [DVD] ERROR 4630 (ref. DVD-3 4.5.2 (2,3,4)) :
ERR_DVD_DSI_ILVU_XX_0
One of the data fields ILVU_EA, NXT_ILVU_SA or NXT_ILVU_SZ is not zero while the ILVU flag is not set.

>>> [DVD] ERROR 4631 (ref. DVD-3 4.5.2 (2)) :
ERR_DVD_DSI_ILVU_EA_ADD
ILVU_EA is different from the relative address of the last pack in this ILVU.

>>> [DVD] ERROR 4632 (ref. DVD-3 4.5.2 (3)) :
ERR_DVD_DSI_ILVU_SA_ADD
NXT_ILVU_SA is different from the relative address of the first pack in the next ILVU with the same VOB_IDN.

>>> [DVD] ERROR 4633 (ref. DVD-3 4.5.2 (3)) :
ERR_DVD_DSI_ILVU_SA_LAST
NXT_ILVU_SA is not 0xFFFFFFFF while this VOBU is part of the last ILVU of this VOB.

>>> [DVD] ERROR 4635 (ref. DVD-3 4.5.2 (4)) :

ERR_DVD_DSI_ILVU_SZ

NXT_ILVU_SZ does not equal the size (as a number of LBs) of the next ILVU with the same VOB_IDN.

>>> [DVD] ERROR 4636 (ref. DVD-3 4.5.2 (4)) :
ERR_DVD_DSI_ILVU_SZ_NONE

NXT_ILVU_SZ is not 0xFFFF while no next ILVU exists.

>>> [DVD] ERROR 4640 (ref. DVD-3 4.5.2 (5..8)) :
ERR_DVD_DSI_VOB_DATA_IDENT

One of the data fields VOB_V_S_PTM, VOB_V_E_PTM, VOB_A_STP_PTM or VOB_A_GAP_LEN is not identical in every VOBU of this VOB.

>>> [DVD] ERROR 4641 (ref. DVD-3 4.5.2 (5)) :
ERR_DVD_DSI_VOB_V_S_PTM

The VOB_V_S_PTM is different from the presentation start time of the first video frame (in display order !) of the first GOP in this VOB.

Recall that video data is always present in the first VOBU of a VOB.

- Checked at the end of the very first VOBU.

>>> [DVD] ERROR 4642 (ref. DVD-3 4.5.2 (6)) :
ERR_DVD_DSI_VOB_V_E_PTM

VOB_V_E_PTM is different from the presentation termination time of the last video frame (in display order !) of the last GOP in this VOB and video data is present in the last VOBU of the VOB.

== VOBU_E_PTM of the last VOBU of the VOB.

- Checked at the end of a VOB.

>>> [DVD] ERROR 4643 (ref. DVD-3 4.5.2 (6)) :
ERR_DVD_DSI_VOB_V_E_PTM_MULT

If no video data is present in the last VOBU of the VOB or the video has stopped earlier, then imaginary video is to be used : VOB_V_E_PTM does not specify a display time on video grid.

- Checked at the end of a VOB.

>>> [DVD] ERROR 4645 (ref. DVD-3 4.5.2) :
ERR_DVD_DSI_VOB_V_STC_OFF

>>> [DVD] ERROR 4651 (ref. DVD-3 4.5.2 (7,8)) :
ERR_DVD_DSI_VOB_A_NON_ILVU

One of the VOB_A_STP_PTM or VOB_A_GAP_LEN data fields is not zero while not in an Interleaved Block.

>>> [DVD] ERROR 4652 (ref. DVD-3 4.5.2 (7,8)) :
ERR_DVD_DSI_VOB_A_NOTPRES_0

One of the VOB_A_STP_PTM or VOB_A_GAP_LEN data fields is not zero for one (of the 8 possible) audio streams which is not present.

>>> [DVD] ERROR 4653 (ref. DVD-3 4.5.2 (7,8)) :
ERR_DVD_DSI_VOB_A_NODISC_0

This error message is generated in the 2 following cases :

1. One of the VOB_A_STP_PTM1 or VOB_A_GAP_LEN1 data fields is not zero while there are no discontinued points.
2. One of the VOB_A_STP_PTM2 or VOB_A_GAP_LEN2 data fields is not zero while there is only one discontinued point.

>>> [DVD] ERROR 4655 (ref. DVD-3 4.5.2 (7)) :
ERR_DVD_DSI_VOB_A_STP_PTM

One of the VOB_A_STP_PTM values is different from the matching stop time of the audio at the discontinuity.

>>> [DVD] ERROR 4656 (ref. DVD-3 4.5.2 (7)) :
ERR_DVD_DSI_VOB_A_STP_PTM_S
This error message is generated in 2 distinct cases :

1. VOB_A_STP_PTM1 is smaller than 40 msec.
2. VOB_A_STP_PTM2 is smaller than (or equal to) VOB_A_STP_PTM1.

>>> [DVD] ERROR 4657 (ref. DVD-3 4.5.2 (7)) :
ERR_DVD_DSI_VOB_A_STP_PTM1
This error message is generated in one of the 2 following cases :

1. The specified VOB_A_STP_PTM1 & VOB_A_GAP_LEN1 combination exceeds the specified video presentation termination time VOB_V_E_PTM :

   VOB_A_STP_PTM1 + VOB_A_GAP_LEN1 + 40ms > VOB_V_E_PTM

2. The specified VOB_A_STP_PTM2 & VOB_A_GAP_LEN2 combination exceeds the specified video presentation termination time VOB_V_E_PTM :

   VOB_A_STP_PTM2 + VOB_A_GAP_LEN2 + 40ms > VOB_V_E_PTM

>>> [DVD] ERROR 4658 (ref. DVD-3 4.5.2 (7)) :
ERR_DVD_DSI_VOB_A_STP_PTM2
The specified VOB_A_STP_PTM1 & VOB_A_GAP_LEN1 combination exceeds the 2nd audio presentation stop time VOB_A_STP_PTM2 :

VOB_A_STP_PTM1 + VOB_A_GAP_LEN1 + 1 sec > VOB_A_STP_PTM2

>>> [DVD] ERROR 4659 (ref. DVD-3 4.5.2 (8)) :
ERR_DVD_DSI_VOB_A_GAP_LEN
The VOB_A_GAP_LEN value is different from the discontinuation period of audio at the discontinued point.

9.3.18.3 SML_AGLI Checks

Observations :
1. An Angle Block (AGL_C_BLK) consists of max. 9 Angles, each composed of exactly 1 Angle Cell (AGL_C).
2. One AGL_C consists of an integer number of ILVUs, of which the start address is described by the DSI SML_AGLI data.
3. The SML_AGLI data describes a sequential ‘slice’ of a complete Angle Block, containing always a part (of identical duration) of each of the 9 possible Angle Cells.
4. Each ILVU consists of an integer number of VOBUs, possible more than 1.

Used Cross Check Parameters :
A dedicated flag "Seamless_Angle_Change_flag" & parameter "Cell_Block_type" have been defined & used for these checks. These match a field with the same name of the PGCI - C_PBI - C_CAT data structure (cf. [DVD-3] Table 4.3.5-1 (1)) and is made available through the Xcheck data file. Also the number of Angles defined in the current Title (defined by the VMGI - TT_SRPT - TT_SRP(i) - AGL_Ns field, cf. [DVD-3] Table 4.1.2-2) is passed through the Xcheck data file.

As a consequence, the checks using any of these parameters can only be properly performed when the proper Cross Checks data (file) is present. If this file is missing, rather than disabling these checks, the necessary parameters otherwise retrieved from this file, are given their default value (which in most cases comes down to a de facto disabling of the checks) :

Parameter	Value	Comment
Seamless_Angle_Change_flag	0 (FALSE)	Non-Seamless Angle Change
Cell_Block_type	1	No Angle Block
AGL_Ns	1	No Angles

>>> [DVD] ERROR 4660 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_NOT0
This error message is generated in 2 distinct cases, when one of the SML_AGL_C[#n]_DSTA entries is not zero and :

1. no Angle Block exists
2. the Angle Block is non-seamless

>>> [DVD] ERROR 4661 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_NR
The n-th SML_AGLI_C[#n]_DSTA entry contains a non-zero value, although there are fewer than n angles defined (by the VMGI.TT_SRPT.TT_SRP(i).AGL_Ns field, cf. [DVD-3] Table 4.1.2-2 (2)).

>>> [DVD] ERROR 4662 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_0
The n-th SML_AGLI_C[#n]_DSTA entry contains a zero value, although a Seamless Angle Change has been indicated defined (by the Seamless_Angle_Change_flag, cf. above).

>>> [DVD] ERROR 4663 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_LOC
The n-th SML_AGLI_C[#n]_DSTA entries AGL_C location flag is not zero, which is the only value allowed.

>>> [DVD] ERROR 4664 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_STRT
The n-th SML_AGLI_C[#n]_DSTA entry AGL_C field specifies an incorrect (non-existing) destination ILVU start address.

>>> [DVD] ERROR 4665 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_LAST
The n-th SML_AGLI_C[#n]_DSTA entry AGL_C field destination ILVU start address has not the mandatory 0x7FFFFFFF value for every DSI in (each VOBU of) the last ILVU of a Cell.

- Checked at the end of an ILVB, by verifying all AGL_C references still stored in the reference list. These should all belong to the last ILVU of a Cell and have their AGL_C destination ILVU start address set to 0x7FFFFFFF.

>>> [DVD] ERROR 4666 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_NLAST
The n-th SML_AGLI_C[#n]_DSTA entry AGL_C field destination ILVU start address has the value 0x7FFFFFFF while this DSI does not belong to a VOBU of the last ILVU of a Cell.

- Checked at the end of an ILVB, by verifying all AGL_C references still stored in the reference list. If these don't have their AGL_C destination ILVU start address set to 0x7FFFFFFF, they should not belong to the last ILVU of a Cell.

>>> [DVD] ERROR 4667 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_SIZE
The n-th SML_AGLI_C[#n]_DSTA entry AGL_C field specifies an incorrect destination ILVU size.

>>> [DVD] ERROR 4668 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_SIZE_0
The n-th SML_AGLI_C[#n]_DSTA entry AGL_C field destination ILVU size is not zero while the ILVU flag is not set.

>>> [DVD] ERROR 4669 (ref. DVD-3 4.5.3) :
ERR_DVD_DSI_SML_AGLI_L_SIZE
The n-th SML_AGLI_C[#n]_DSTA entry AGL_C field destination ILVU size is not zero for the last ILVU of a Cell.

- Checked at the end of an ILVB, by verifying all AGL_C references still stored in the reference list. These should all belong to the last ILVU of a Cell and have their AGL_C destination ILVU size set to zero.

9.3.18.4 VOBU_SRI Checks

- The following VOBU_SRI checks are used, unless specified explicitly otherwise, for both FWD and BWD versions; This is referred to below as "F/BWD".
- Furthermore, the (error) messages may result from any of the 42 VOBU_SRI table F/BWDI entries, unless a specific entry, e.g. "F/BWDI Video" , " FWDI Next" or "BWDI Prev", is specified.
- The term "predecessor" for VOBU_SRI entry (i) is interpreted as the VOBU matching the VOBU_SRI table entry (i-1). In other words, the VOBU one entry closer to the current VOBU.
- The verification of the VOBU_SRI table entries is done separately for forward & backward references :
    - backward references (BWDI) are checked immediately using a "VOBU" list with all the VOBUs in the part of the Cell preceding the current VOBU. This entry contains all necessary information for the checks (relative address, start time, video flag, etc.).

This list is a generic list with VOB scope generated & maintained by the *vob_verf* object.

- forward references (FWDI) are checked whenever one VOBU referenced by a DSI in the preceding part of the Cell is encountered. Therefore two "ref" lists are used containing all forward VOBU_SRI references not checked yet : one to verify references to existing VOBUs; the second one to verify missing references.

Since all these references are restricted to the current Cell, these "ref" lists are generated during the current Cell parsing, valid for the current Cell only and destroyed at the end of a Cell.

>>> [DVD] ERROR 4671 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWD_EX_1
The V_FWD_Exist1 flag specifies incorrectly (non-)existing video data in the destination VOBU.

- Checked at VOBU end in case of forward reference FWDI.

>>> [DVD] ERROR 4672 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWD_EX_1_0
The V_FWD_Exist1 flag is not zero while the destination VOBU does not exist.

- Checked immediately also in case of forward reference FWDI.

>>> [DVD] ERROR 4673 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWD_EX_2
This error message is generated in 2 distinct cases :

When the V_FWD_Exist2 flag specifies incorrectly (non-)existing video data between the VOBU to be presented just after/before the predecessor and

1. the VOBU to be presented just before/after VOBU addressed by FWDA[#n].
2. the last/first VOBU in the Cell, in case the VOBU addressed by FWDA[#n] does not exist.

- Checked at VOBU and Cell end in case of forward reference FWDI.

>>> [DVD] ERROR 4674 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWD_EX_2_0
The V_FWD_Exist2 flag is not zero while both the destination VOBU and the predecessor do not exist.

- Checked at Cell end in case of forward reference FWDI.

>>> [DVD] ERROR 4675 (ref. DVD-3 4.5.4 (1,4)) :

ERR_DVD_DSI_SRI_FBWD_EX_2_1

The V_FWD_Exist2 flag is not zero for one of the VOBU_SRI entries FWDI 1 to FWDI 15.

- Checked immediately, also in case of forward reference FWDI.

>>> [DVD] ERROR 4677 (ref. DVD-3 4.5.4 (2,5)) :
ERR_DVD_DSI_SRI_FBWD_EX_1_FLST

The FWDI Next V_FWD_Exist1 / BWDI Prev V_BWD_Exist1 flag is not zero for the last/first VOBU of a Cell.

- Checked at Cell end in case of forward reference FWDI.

>>> [DVD] ERROR 4678 (ref. DVD-3 4.5.4) :
ERR_DVD_DSI_SRI_FBWD_EX_1_V

DSI_SRI: The V_FWD_Exist1 or V_BWD_Exist1 flag is incorrectly specified.

>>> [DVD] ERROR 4679 (ref. DVD-3 4.5.4) :
ERR_DVD_DSI_SRI_FBWD_EX_2_V

DSI_SRI: The V_FWD_Exist2 or V_BWD_Exist2 flag is incorrectly specified.

>>> [DVD] ERROR 4680 (ref. DVD-3 4.5.4) :
ERR_DVD_DSI_SRI_FBWDA_ILL

FWDA specifies an illegal VOBU relative start address. More specifically a sector address is specified which is not a NV_PCK marking the start of a VOBU.

- Checked at VOBU end in case of forward reference FWDI.

>>> [DVD] ERROR 4681 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWDA_ERR

FWDA specifies an incorrect VOBU relative start address.

- Here "relative address" is the packs count from the start of the current VOBU, but within the current Cell! Note that, when the reference exceeds the cell boundaries it is considered incorrect.
- Checked at VOBU end in case of forward reference FWDI.

>>> [DVD] ERROR 4682 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWDA_S_PTM

FWDA specifies a VOBU that is not being presented (n x 0.5 sec) after/before the current VOBU presentation start time, as is required for FWDI entry #n.

In principle is required (for forward references) :

$$\text{VOBU\_S\_PTM}_{\text{target VOBU}} = \text{VOBU\_S\_PTM}_{\text{current VOBU}} + (n \cdot 0.5\text{sec})$$

In practice this becomes :

$$\text{VOBU\_S\_PTM}_{\text{target VOBU}} \leq \text{VOBU\_S\_PTM}_{\text{current VOBU}} + (n \cdot 0.5\text{sec})$$

$$\text{VOBU\_E\_PTM}_{\text{target VOBU}} > \text{VOBU\_S\_PTM}_{\text{current VOBU}} + (n \cdot 0.5\text{sec})$$

For backward references the plus sign has to be replaced by a minus sign.

- Checked at VOBU end in case of forward reference FWDI.

>>> [DVD] ERROR 4683 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWDA_ERRPOS

FWDA specifies a VOBU start address after/before the current Cell end/start.

(This is simply verified by comparing the F/BWDA value with the relative address of the current VOBU within the current Cell.)

- Checked at Cell end in case of forward reference FWDI.

>>> [DVD] ERROR 4684 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWDA_EXST
FWDA specifies an existing destination VOBU, i.e. start address different from 0x3FFFFFFF, while the target VOBU does not exist.

- Checked at VOBU end in case of forward reference FWDI.

>>> [DVD] ERROR 4685 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWDA_N_EXST
FWDA specifies a non-existing destination VOBU, i.e. start address equals 0x3FFFFFFF, while the target VOBU does exist.

- Checked at VOBU and Cell end in case of forward reference FWDI.

>>> [DVD] ERROR 4686 (ref. DVD-3 4.5.4 (1,4)) :
ERR_DVD_DSI_SRI_FBWDA_TIMEX
BWDA[#n] specifies a VOBU start address different from 0x3FFFFFFF while the relevant time exceeds the Cell's presentation start time.

! This is not checked for FWDA[#n], since the Cell presentation termination time is not known in advance.

>>> [DVD] ERROR 4687 (ref. DVD-3 4.5.4) :
ERR_DVD_DSI_SRI_FBWDA_EXST_FLST
FWDI Next FWDAn is not 0x3FFFFFFF for the last VOBU of a Cell, or

BWDI Prev BWDAn and BWDI Video BWDAn is not 0x3FFFFFFF for the first VOBU of a Cell.

- Checked at Cell end in case of forward reference FWDI.

>>> [DVD] ERROR 4689 (ref. DVD-3 4.5.4) :
ERR_DVD_DSI_SRI_FBWD_NOPRED
No predecessor was found although expected to be present !

- This is not a real DVD specification violation, but rather a system message.

9.3.18.5 SYNCI Checks

Remarks :

- The following SYNCI checks are used for both A_SYNCA and SP_SYNCA table entries, unless specified otherwise.
- The verification of the SYNCI table entries is done separately for forward & backward references :
  - backward references are checked immediately using a "pack" list with all the packs in the part of the VOB preceding the current VOBU. This entry contains all necessary information for the checks (relative address, start time, etc.).
  - forward references are checked whenever a VOBU referenced by the current DSI in the preceding part of the VOB is encountered. Therefore a "ref" list, sorted in ascending order on the absolute pack address, is used containing all forward SYNCI references not yet checked.

Of both lists, there are two instances : one for Audio packs (A_PCK), one for Sub-picture packs (SP_PCK).

Since all these references are restricted to the current VOB, these lists are generated during the current VOB parsing, valid for the current VOB only and destroyed at the end of a VOB.

To verify the presentation start time of the target A/SP_PCK specified by A/SP_PCKA, the above mentioned pack list is extended with the presentation start time of each pack as defined by the specification.

- The constraint :

When there are less than 32 SP streams, the SP_PCK location flag and SP_PCKA address should be zero.

has precedence over the constraint :

When SP_PCK location flag == 0, then the SP_PCKA address should have all its bits set.

- When SP_PCKs are located AFTER the current NV_PCK, no reference to them is necessary for correct DVD data handling (in this case the SP_PCK location flag is set to zero and SP_PCKA has all its bits set). So, no forward sub-picture references have to be verified and no list is needed to store them.
- The SP_PCK location flag has two meanings :
  1. Specify the target SP_PCK location w.r.t. the current NV_PCK
  2. Indicate there is no target SP_PCK (all bits of SP_PCKA are set)
- Interpretation of combined SP_PCK location flag and SP_PCKA values :

  SP_PCKA should not be filled in when :
  1. The SP_PCK location flag == 0 ➔ It should have all its bits set.
  2. The VOBU specified by SP_PCKA contains no video ➔ It is not clear what value SP_PCKA then should contain, but all zero is assumed.

Used Cross Check Parameters :
For verification, the number of audio and sub-picture streams present in the VOB is needed. These values can not directly be retrieved from the VOB data but are only available in the DVD VTSI data structure : specified by resp. VTSI .VTSI_MAT .VTS_AST_Ns and .VTS_SPST_Ns (cf. [DVD-3] 4.2.1 RBP 514 and 596). They are passed by the Xcheck data file.

As a consequence, the checks using one of these parameters can only be properly performed when the proper Cross Checks data (file) is present. If this file is missing, rather than disabling these checks, the necessary parameters otherwise retrieved from this file, are given their default value (which in case of Sub-picture streams comes down to a de facto disabling of the checks) :

Parameter	Default Value	Comment
AST_Ns	The dvd_core VTS_AST_NS field	Indirectly retrieved from the script file settings
SPST_Ns	0	No Sub-pictures

>>> [DVD] ERROR 4690 (ref. DVD-3 4.5.5 (1,2)) :
ERR_DVD_DSI_SYNCI_PCK_ADD
The specified relative address does not match an existing A/SP_PCK.

- Here "relative address" is the packs count from the start of the current NV_PCK.
- Generated when a search scan of the VOBU absolute address list for a VOBU matching the specified relative address A/SP_PCKA fails.
- For backward references, this message is generated when a search scan of the pack absolute address list for a pack matching the specified relative address A/SP_PCKA fails.
- To check forward references, the current pack address is continuously compared with that of the first entry of the pack “ref” list. When the first address is beyond the second one, i.e. the current stream position has passed the position of the first unchecked reference, this message is generated.

>>> [DVD] ERROR 4691 (ref. DVD-3 4.5.5 (1,2)) :
ERR_DVD_DSI_SYNCI_PCK_TIME
The target A/SP_PCK contains an AU with a presentation time which is not simultaneously with nor immediately after the current VOBU presentation start time.

- This AU consists of an audio frame in case of an A_PCK, or the first packet of the first SPU in case of a SP.
- Only when the target VOBU is found in the VOBU address list, the presentation start time stored in the list is compared with that of the current VOBU. The first should then be properly set to the PTS of the appropriate AU, resp. the first audio frame or first SPU packet.
- The term “immediately after” is interpreted as described for the PCI.NSML_AGLI Destination address of AGL_C[#n] (cf. [DVD-3] 4.4.2) : the closest later PTM than the current VOBU ’s PTM.

>>> [DVD] ERROR 4692 (ref. DVD-3 4.5.5 (1,2)) :
ERR_DVD_DSI_SYNCI_PCK_STRM
The A/SP_PCK location and A/SP_PCKA fields should be zero for non-present audio or sub-picture streams.

- When the relevant Cross Check parameters are not available, the checks are performed using parameter default values (which are logged to the verifier report).

>>> [DVD] ERROR 4693 (ref. DVD-3 4.5.5) :
ERR_DVD_DSI_SYNCI_PCK_EXST
The N-th A/SP_PCKA field is zero, indicating this audio or sub-picture stream does not exist, while it does, because there are more than N streams present.

>>> [DVD] ERROR 4694 (ref. DVD-3 4.5.5 (1)) :
ERR_DVD_DSI_SYNCI_A_PCK_LOC0
The A_PCK location flag is set while the target A_PCK does not exist.

- A_PCK location flag is not zero, while A_PCKA is.
- This is not explicitly stated in the spec and therefore is implemented as an ODDITY.

>>> [DVD] ERROR 4696 (ref. DVD-3 4.5.5 (1)) :
ERR_DVD_DSI_SYNCI_A_PCKA_EX
The A_PCKA target address does not have all bits set while the address value exceeds the maximum value.

- The target A_PCK is searched for in the VOBU list using the presentation start time of the current VOBU. When found, its relative address is checked to be within 15bit range.

>>> [DVD] ERROR 4697 (ref. DVD-3 4.5.5 (2)) :
ERR_DVD_DSI_SYNCI_SP_PCK_LOC0
The SP_PCK location flag is set while the target SP_PCK does not exist.

- SP_PCK location flag is not zero, while SP_PCKA is all zero or all 1.

>>> [DVD] ERROR 4699 (ref. DVD-3 4.5.5 (2)) :
ERR_DVD_DSI_SYNCI_SP_PCKA_V
The SP_PCK location flag is set but the SP_PCKA target VOBU does not contain the video data.

- This is checked by looking up the VOBU referenced by SP_PCKA in the current VOB's VOBU list and check its video flag.

>>> [DVD] ERROR 4700 (ref. DVD-3 4.5.5 (2)) :
ERR_DVD_DSI_SYNCI_SP_NO_PCK_TIME
SYNCI SP_SYNCA 'number' : No target SP_PCK found with a presentation start time which is simultaneously with, nor immediately after the matching VOBU presentation start time.

>>> [DVD] ERROR 4701 (ref. DVD-3 4.5.5 (1)) :
ERR_DVD_DSI_SYNCI_A_PCK_TIME
SYNCI A_SYNCA 'number' : The target A_PCK has a presentation start time 'value' which is not simultaneously with, nor immediately after the matching VOBU presentation start time 'value' (VOBU_S_PTM at larger distance than 1 frame duration time.

>>> [DVD] ERROR 4702 (ref. DVD-3 4.5.5 (2)) :
ERR_DVD_DSI_SYNCI_SP_PCK_TIME
SYNCI SP_SYNCA 'number' : The target SP_PCK has a presentation start time 'value' which is not simultaneously with, nor immediately after the matching VOBU presentation start time.

>>> [DVD] ERROR 4703 (ref. DVD-3 4.5.5 (2)) :
ERR_DVD_DSI_SYNCI_SP_NO_VOBU
SYNCI SP_SYNCA 'number' : The specified SP_PCKA address 'address' does not match the NV_PCK's address of a VOBU including any SP-PCK.

9.3.19 DVD NCMD checks

>>> [DVD] ERROR 4801 (ref. DVD-3 4.6.3.2) :
ERR_DVD_NCMD_RES_ILL
The reserved field from Command_Type1 shall be '0'

>>> [DVD] ERROR 4802 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_OPERAND_RES_ILL
The reserved field from the Command's Operand should be '0'

>>> [DVD] ERROR 4803 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_RES_CMD
Illegal Command specified!!!

>>> [DVD] ERROR 4804 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_ARG_RES_ILL
The reserved field from this Command's Argument should be '0'

>>> [DVD] ERROR 4805 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_ARG_RES_VALUE
A field from the Command's argument specifies a reserved value.

>>> [DVD] ERROR 4810 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_DOMAIL_ILL1
A Command is found in an illegal Domain .

>>> [DVD] ERROR 4811 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_DOMAIN_ILL2
A Command is found in an illegal Domain.

>>> [DVD] ERROR 4812 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_AREA_ILL1
A Command is found in an illegal Area.

>>> [DVD] ERROR 4813 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_AREA_ILL2
A Command is found in an illegal Area.

>>> [DVD] ERROR 4814 (ref. DVD-3 4.6.4.1) :
ERR_DVD_NCMD_GOTO_ILL
A Command specified an illegal Command number.

>>> [DVD] ERROR 4815 (ref. DVD-3 4.6.4.1) :
ERR_DVD_NCMD_ILL_GMPGOTO
An Operand was used which may not be used in this command.

>>> [DVD] ERROR 4816 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_PTL_ILL
A Command specified an illegal Parental_Level.

>>> [DVD] ERROR 4820 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_HLBTNN_ILL
A Command specified an illegal HL_BTNN.
>>> [DVD] ERROR 4821 (ref. DVD-3 4.6.4.2) :

ERR_DVD_ NCMD_LINK_HLBTNN_ILL2
A Command specified an illegal HL_BTNN.

>>> [DVD] ERROR 4822 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_S_RESV
A Reserved Link_sub_instruction was found.

>>> [DVD] ERROR 4823 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_PGCN_ILL
A Command specified an illegal PGCN.

>>> [DVD] ERROR 4824 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_PGN_ILL
A Command specified an illegal PGN.

>>> [DVD] ERROR 4825 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_X_PGN
A Command specified a non-existing PGN.

>>> [DVD] ERROR 4826 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_PTTN_ILL
A Command specified an illegal PTTN.

>>> [DVD] ERROR 4827 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_CN_ILL
A Command specified an illegal CN.

>>> [DVD] ERROR 4828 (ref. DVD-3 4.6.4.2) :
ERR_DVD_NCMD_LINK_X_CN
A Command specified a non-existing CN.

>>> [DVD] ERROR 4835 (ref. DVD-3 4.6.4.3) :
ERR_DVD_NCMD_JUMP_CN_ILL
A Command specified an illegal CN.

>>> [DVD] ERROR 4836 (ref. DVD-3 4.6.4.3) :
ERR_DVD_ NCMD_JUMP_TTN_ILL
A Command specified an illegal TTN.

>>> [DVD] ERROR 4837 (ref. DVD-3 4.6.4.3) :
ERR_DVD_ NCMD_JUMP_VTSTTN_ILL
A Command specified an illegal VTS_TTN.

>>> [DVD] ERROR 4838 (ref. DVD-3 4.6.4.3) :
ERR_DVD_ NCMD_JUMP_VTSN_ILL
A Command specified an illegal VTSN.

>>> [DVD] ERROR 4839 (ref. DVD-3 4.6.4.3) :
ERR_DVD_ NCMD_JUMP_VTSN_VTS_SPACE
A Command specified an illegal VTS.

>>> [DVD] ERROR 4840 (ref. DVD-3 4.6.4.3) :
ERR_DVD_ NCMD_JUMP_VMGM_PGCN_ILL
A Command specified an illegal PGCN.

>>> [DVD] ERROR 4841 (ref. DVD-3 4.6.4.3) :

ERR_DVD_NCMD_JUMP_DOAMINID
The specified Domain_ID is illegal for the Menu_ID.

>>> [DVD] ERROR 4842 (ref. DVD-3 4.6.4.3) :
ERR_DVD_NCMD_VMGMPGCN_DOMID
The specified VMGM_PGCN should be '0x0000' for the Domain_ID.

>>> [DVD] ERROR 4850 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_SETSYS_PARAM
The Parameter_Number exceeds the allowed maximum.

>>> [DVD] ERROR 4851 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_SETSYS_Ns
If the I_flag for SetSystem Instruction is '0':
1) If the A_flag is '0' then ASTN shall be '0',
2) If the SP_flag is '0' then SPSTN shall be '0',
3) If the AGL_flag is '0' then AGLN shall be '0'

>>> [DVD] ERROR 4852 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_SETSYS_Ns_MAX
The ASTN, SPSTN or AGLN is illegal with the used flag.

>>> [DVD] ERROR 4853 (ref. DVD-3 4.6.4) :
ERR_DVD_NCMD_GEN_PARAM
The General_Parameter_Number should be maximum 15 for this PRM_Flag.

>>> [DVD] ERROR 4854 (ref. DVD-3 4.6.4) :
ERR_DVD_ NCMD_SYS_PARAM
The System_Parameter_Number should be maximum 23 for this PRM_Flag.

>>> [DVD] ERROR 4855 (ref. DVD-3 4.6.4.5) :
ERR_DVD_ NCMD_SYS_PARAM_ILL_CMD
No System_Parameter_Number should be specified for this Set_Field.

>>> [DVD] ERROR 4860 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_IMN_VALUE_ILL
No Immediate_Value should be specified for this Set_Field.

>>> [DVD] ERROR 4861 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_GPRMN_ILL
ASTN GPRMN should be '0' when the A_flag is '0'.
SPSTN GPRMN should be '0' when the SP _flag is '0'.
AGLN GPRMN should be '0' when the AGL_flag is '0'.

>>> [DVD] ERROR 4862 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_SPDISPFLAG
The SP_Disp_Flag should be '0' when the SP_Flag is '0'.

>>> [DVD] ERROR 4863 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_HL_BTNN_NULL
The HL_BTNN value should not be when specified in this command.

>>> [DVD] ERROR 4864 (ref. DVD-3 4.6.4.5) :
ERR_DVD_NCMD_HL_BTNN_LARGE
The HL_BTNN should maximum be 36.

>>> [DVD] SYNTAX ERROR 4870 (ref. DVD-3 4.3.3-1) :

ERR_DVD_PGC_CMD_LARGE
PGC_CMDT: The number of combined commands is larger then allowed. Only 128 PRE_CMDs are parsed!!!

9.3.20 DVD Sector checks

The checks, which comprise the verification of a sector, are all based on the data sector as defined in [DVD] 3.2.

>>> [DVD] ERROR 4951 (ref. DVD-1 3.1.4) :
ERR_SECTOR_NUMBER
The physical sector number must correspond to the logical sector number. Depending on the type of disc (Single/Dual layer, Parallel/Opposite track) a relation between the logical sector number and physical sector number is verified. The error notification contains the values of the logical and physical sector number.

>>> [DVD] ERROR 4952 (ref. DVD-1 3.2.2) :
ERR_SECTOR_ID
The reserved field of the Sector information (of the Identification Data) must be zero.

>>> [DVD] ERROR 4953 (ref. DVD-1 3.2.3) :
ERR_SECTOR_IED
This is a CRC check. The CRC value calculated over the Identification Data must be equal to the value of the ID Error Detection Code. The error notification contains the value of the calculated remainder.

>>> [DVD] ERROR 4954 (ref. DVD-1 3.2.4) :
ERR_SECTOR_CPR_MAI
The following requirements must be met:

- If the data sector is part of the lead-in area and the relative sector number is at least 2 and at most 15, the CPR_MAI must be according to [DVD] 3.2.4.1. This is the contents provider information. Three aspects must be verified: First, the value of CPS_TY must be either zero or one. Second, the CPR_MAI contains four reserved bytes. Finally, the in this item discussed sectors must have the same values for the CPS_TY and RMA attributes.
- If the data sector is part of the lead-in area and the relative sector number is smaller than 2 or larger than 15, the CPR_MAI must be set to '0' in all bits.
- If the data sector is part of the data area, the CPR_MAI must be according to [DVD] 3.2.4.2. The CPR_MAI contains some copyright information, but also some reserved fields. These reserved fields must contain the value zero.
- If the data sector is part of the middle or lead-out area, the CPR_MAI must be set to '0' in all bits.

>>> [DVD] ERROR 4955 (ref. DVD-1 3.2.4 .1) :
ERR_SECTOR_CPR_RMA
An oddity message is generated when the disc is not allowed to be played in any region. For a lead-in sector with relative sector number in [2..15], not all the RMA values may be one.

>>> [DVD] ERROR 4956 (ref. DVD-1 3.2.5) :
ERR_SECTOR_EDC
This is a CRC check. The CRC value of the Data Sector, without the EDC value, must be equal to the EDC value. The error notification contains the value of the calculated remainder.

9.3.21 Filesystem checks

9.3.21.1 UDF Filesystem checks

>>> [DVD] ERROR 5001 (ref. DVD-2 2.1 2, ECMA 3/8.1.2) :
ERR_FSYS_WRONG_SECTOR_SIZE
The logical sector size must be a multiple of 512 bytes. Furthermore, according to [UDF], the sector size must be 2048.

>>> [DVD] ERROR 5002 :
ERR_FSYS_WRONG_DISC_TYPE
Wrong or unknown disc (image) type.

>>> [DVD] ERROR 5003 (ref. ECMA 3/8.1.3) :
ERR_FSYS_NLOGICAL_SECTOR_TOO_SMALL
Logical sector numbers shall be consecutive integers in ascending order. The smallest logical sector number of a volume shall be 0, the largest shall be greater than 256.

>>> [DVD] ERROR 5004 :
ERR_FSYS_NO_FILE
Could not open/read disc (image).

>>> [DVD] ERROR 5005 :
ERR_FSYS_SECTOR_NOT_FOUND
The addressed sector could not be found.

>>> [DVD] ERROR 5006 :
ERR_FSYS_SECTOR_NOT_READ
The addressed sector could not be read.

9.3.21.1.1 DVD Filesystem ECMA1 checks

The checks which comprise the verification of the [UDF] file system are listed according to the following scheme: The required checks are presented by following the [ECMA] standard. The standard is partitioned into four parts and the required checks are presented analogous to this standard.

Occasionally, checks refer to DVD-2. The verifier for the DVD file system will be initialised with a Boolean indicating whether or not a DVD-2 disk will be verifier. Checks which should only be performed for DVD-2 disks should be guarded using this initially passed value.

The DVD file system is based on three standards. The basis is the [ECMA] standard. On top of this there is the [UDF] standard. Every definition that complies to [UDF] also complies to [ECMA]. The [UDF] standard is a ‘smaller’ definition. Finally, the [DVD] standard is defined on top of the [UDF] standard. Every check that needs to be performed originates from one (or more) of these standards. If a certain requirement is stated in more than one standard, a reference to the most generic standard is given. For example, if a requirement is stated in both the [UDF] and [DVD] standard, the reference to the [UDF] standard is presented.

Shaded checks are not yet implemented. To implement these checks a lot of time is required, and these checks are not very interesting with respect to [DVD] file systems. The checks mostly concern parts of the [ECMA] standard which are not used by the [DVD] standard.

>>> [DVD] ERROR 5051 :
ERR_FSYS_FLAGS_NOT_NULL
Reserved flags shall be ZERO.

>>> [DVD] ERROR 5052 (ref. ECMA 3/8.4.4) :
ERR_FSYS_REMAINDER_NOT_NULL
All space after the end of the last descriptor up to the end of the logical sector shall be recorded as all #00 bytes.

>>> [DVD] ERROR 5053 :
ERR_FSYS_RESERVED_NOT_NULL
Reserved bytes not #00

>>> [DVD] ERROR 5054 :
ERR_FSYS_EXTENT_TOO_SMALL
Structure overflows extent.

>>> [DVD] ERROR 5055 :
ERR_FSYS_EXTENTS_OVERLAP
Sectors of extents shall not overlap.

>>> [DVD] ERROR 5060 (ref. ECMA 7.2, UDF 2.1.1) :
ERR_DSTRING_INVALID_CHAR
The characters must comply to the Unicode 1.1 specification.

>>> [DVD] ERROR 5061 (ref. ECMA 7.2, UDF 2.1.1) :
ERR_DSTRING_INVALID_COMPRESSIONID
Two compression algorithms are supported: the value of CompressionID must be either eight or sixteen.

This is checked by the parser.

>>> [DVD] ERROR 5062 (ref. ECMA 1/7.3) :
ERR_TIMESTAMP_OUT_OF_RANGE
Each of the fields must comply to the corresponding interval stated in ECMA 1/7.3:

- Year: 1..9999.
- Month: 1..12.
- Day: 1..31.
- Hour: 0..23.
- Minute: 0..59.
- Second: If the value of Type equals 2 than 0..60, otherwise 0..59.
- Centiseconds: 0..99.
- Hundreds Of Microseconds: 0..99.
- Microseconds: 0..99.

>>> [DVD] ERROR 5063 (ref. UDF 2.1.4.1, ECMA 1/7.3) :
ERR_TIMESTAMP_NOT_LOCAL_TIME
The type of the time stamp must equal the value one.

>>> [DVD] ERROR 5064 (ref. ECMA 1/7.4, UDF 2.1.5) :
ERR_REGID_IDENTIFIER_NOT_COMPLIANT
The value of the identifier must comply to the values specified in [UDF] 6.2, it must contain one of the byte sequences listed in the table.

>>> [DVD] ERROR 5065 (ref. UDF 6.3 and 2.1.5.3, ECMA 1/7.4) :
ERR_REGID_OS_NOT_RECOGNIZED
The values for the OS class and OS identifier must comply to the values specified in the table of section [UDF] 6.3.

>>> [DVD] ODDITY 5066 (ref. UDF 2.1.5.3, ECMA 1/7.4) :
ERR_REGID_UDF_REVISION
The value of the UDF revision in the Domain Identifier Suffix or the UDF Identifier Suffix must be 0x0102.

>>> [DVD] ODDITY 5067 (ref. UDF 6.3) :
ERR_REGID_IDENTIFIER_NOT_CONSISTENT
Regid implementation identifier shall be consistent with Primary Volume Descriptor implementation identifier.

>>> [DVD] ERROR 5068 (ref. UDF 2.1.3, ECMA 1/7.2.12) :
ERR_DSTRING_LENGTH_TOO_LARGE
For fixed length character fields of length n, the n^{th} byte contains the length of the string. This value may not be longer than the maximum length of the character field.

>>> [DVD] ERROR 5069 (ref. UDF 2.1.3, ECMA 1/7.2.12) :
ERR_DSTRING_REMAINING_NOT_ZERO
If the character field length recorded in the last byte is zero, the other fields must contain the value 0x00. Also, remaining byte positions shall be set to zero.

>>> [DVD] ERROR 5070 (ref. UDF 2.1.2, ECMA 1/7.2) :
ERR_DSTRING_CHARACTER_SET
The value for the CharactersSetType must be 0 and the value for CharacterSetInfo[63] must equal the ASCII string "OSTA Compressed Unicode".

>>> [DVD] ERROR 5080 (ref. DVD-2 2.6.5 and 2.6.7, ECMA 1/7.4) :
ERR_REGID_ NOT_PROTECTED
In case of DVD-2, the value of the flags (of the entity identifier) must be two. Furthermore, the Hard Write-Protect flag and the Soft Write-Protect flag must be set to one.

>>> [DVD] ERROR 5081 (ref. DVD-2 A.2, A.3) :
ERR_REGID_ OS_DEFINED
OS class, identifier, should be 0

9.3.21.1.2 DVD Filesystem ECMA2 checks

>>> [DVD] ERROR 5151 (ref. ECMA 2/8.3.1) :
ERR_VRA_TERMINATOR
The descriptor following a TEA01 (Terminating Extended Area) sequence can only be a BEA01 (Beginning Extended Area) descriptor. Furthermore, a BEA01 descriptor may only follow a BEA01 descriptor or a TEA01 descriptor.

>>> [DVD] ERROR 5152 (ref. ECMA 2/8, 2/9, 3/9) :
ERR_VRA_TYPE_VERSION
The value of the structure type of all the descriptors which can be part of the volume recognition area, must be zero. Furthermore, the value of the structure version field of the BOOT2, BEA01, NSR02, and the TEA01 must be one.

>>> [DVD] ERROR 5153 (ref. ECMA 2/9.1.2) :
ERR_VRA_IDENTIFIER_UNKNOWN
Each descriptor which is part of the volume recognition area has a standard identifier. This identifier must comply to the values presented in the table of [ECMA] 1/9.1.2. These values are "BEA01", "BOOT2", "CD001", "CDW02", "NSR02", and "TEA01".

>>> [DVD] ERROR 5154 (ref. ECMA 2/9.4) :
ERR_VRA_BOOT_EXTENT
The Boot Extent Location and the Boot Extent Length of a boot descriptor need to fit into a volume. However, if the erase flag is set, other boot descriptors may override the descriptor.

9.3.21.1.3 DVD Filesystem ECMA3 checks

>>> [DVD] ERROR 5155 (ref. ECMA 3/3.1) :
ERR_VRA_NSR02_NOT_FOUND
The NSR02 descriptor is not recorded.

>>> [DVD] ERROR 5160 (ref. DVD-2 A.11) :
ERR_VRA_BOOT_DESCRIPTOR
No BOOT descriptor allowed in DVD-2.

>>> [DVD] ERROR 5201 (ref. UDF 2.2.1.2) :
ERR_FSYS_DESCRIPTOR_LENGTH_INCORRECT
Descriptor length is incorrect.

>>> [DVD] ERROR 5202 (ref. ECMA 3/10.5.8) :
ERR_EXTENT_AD_OUT_OF_RANGE
The partition (the combination of position and length) does not fit in the volume.

>>> [DVD] ERROR 5203 (ref. ECMA 3/7.1.1) :
ERR_EXTENT_AD_LENGTH_NOT_MULTIPLE
The extent length of an extent descriptor must be a multiple of the sector size.

>>> [DVD] ERROR 5204 (ref. ECMA 3/7.2.1, 4/7.2.1) :
ERR_TAG_IDENTIFIER_UNKNOWN
The value of the identifier of a tag (of a descriptor) must be in the interval [1..9] for [ECMA] 3/7.2. or in the interval [256.. 265] for [ECMA] 4/7.2.

>>> [DVD] ERROR 5205 (ref. ECMA 3/7.2.2, 4/7.2.1) :
ERR_TAG_VERSION_NOT_TWO
The Descriptor Version must be two.

>>> [DVD] ERROR 5206 (ref. ECMA 3/7.2.3, 4/7.2.1) :
ERR_TAG_CHECKSUM
The Tag Checksum of the tag must be equal to the sum of all the bytes comprising the tag with the exception of the Tag Checksum.

>>> [DVD] ERROR 5207 (ref. ECMA 3/7.2.6, 4/7.2.1) :
ERR_TAG_CRC
The Descriptor CRC of the tag contains the remainder of a CRC calculation. The CRC calculation is performed over a sequence of bytes of length Descriptor CRC Length starting at the first byte after the tag.

>>> [DVD] ERROR 5208 (ref. ECMA 3/7.2.8, 4/7.2.1) :
ERR_TAG_LOCATION
The value of the Tag Location must equal the logical sector containing the first byte of the descriptor.

>>> [DVD] ERROR 5209 (ref. DVD-2 1.5.6, ECMA 3/7.2.5) :
ERR_TAG_SERIAL_NUMBER
The Tag Serial Number must be equal to zero.

>>> [DVD] ERROR 5210 (ref. UDF 2.3, ECMA 3/8.4.2) :
ERR_VDS_NPREVAILING
According to UDF 2, the following requirements exists:

- The number of prevailing PVDs must be one.
- The number of prevailing partition descriptors must be at most two.
- The number of LVDs must be exactly one.
- The number of prevailing USDs must be one.
- The number of prevailing LVIDs must be at least one.

According to [DVD 2.3], the number of prevailing partition descriptors must be one.

>>> [DVD] ERROR 5211 (ref. ECMA 3/8.4.1) :
ERR_VDS_DESCRIPTORS_NOT_IDENTICAL
Every two descriptors which have the same Volume Descriptor Sequence Number in a volume descriptor sequence, must be equal.

>>> [DVD] ERROR 5212 (ref. ECMA 3/8.3, 3/8.4.2) :
ERR_VDS_DESCRIPTORS_TYPE_INVALID
Descriptor tag identifier is not in {1,3,4,5,6,7,8}

>>> [DVD] ERROR 5213 (ref. DVD-2 2.3, UDF 2 and ECMA 3/8.4.2.1) :
ERR_ANCHOR_POINTS_NOT_TWO
According to [UDF] 2, the number of recorded AVDP must be exactly two. They must be placed at two of the following three places: logical sector number 256, N-256, or N, where N is the last addressable sector of a volume.

According to [DVD] 2.3, the AVDP must be recorded at 256 and N, where N is again the last logical sector number.

>>> [DVD] ERROR 5214 (ref. UDF 2, 2.2.3 ECMA 3/10.2) :
ERR_ANCHOR_NO_RESERVE
An AVDP must have a reserve VDS.

>>> [DVD] ERROR 5215 (ref. UDF 2.2.3.1, ECMA 3/10.2) :
ERR_ANCHOR_EXTENTS_TOO_SMALL
The extents identified by an AVDP must be at least 16 logical sectors in size.

>>> [DVD] ERROR 5216 (ref. ECMA 3/8.4.2, 8.4.2.3) :
ERR_VDS_NOT_EQUIVALENT
The main and reserve VDS must be equivalent. This means that they must specify equivalent sets of volume descriptors. The canonical forms must be the same.

>>> [DVD] ERROR 5217 (ref. DVD-2 2.1.7, ECMA 3/8.4.2) :
ERR_VDS_NO_TERMINATOR
According to [DVD] 2.1 (item 7), a terminating descriptor must be used to terminate the VDS.

>>> [DVD] ERROR 5220 (ref. ECMA 3/8.6, 10.1.6) :
ERR_PVD_VOLUME_SEQUENCE_NUMBER
The Volume Sequence Numbers of prevailing PVDs must be numbered in ascending order, starting at one.

>>> [DVD] ERROR 5221 (ref. ECMA 3/8.8.2, 10.10.3, UDF 2) :
ERR_LVID_NOT_CLOSED
The prevailing LVID must be closed, the value of IntegrityType must be one.

>>> [DVD] ERROR 5222 (ref. DVD-2 2.1 7 ECMA 3/8.8.2) :
ERR_LVIS_NO_TERMINATOR
In case of DVD-2 ([DVD] 2.1, item 7), a LVIS must be terminated by a Terminating Descriptor.

>>> [DVD] ERROR 5223 (ref. DVD-2 2.1 8. ECMA 3/8.8.2) :
ERR_LVIS_OPEN_DESCRIPTOR
In case of DVD-2 ([DVD] 2.1, item 8), a LVIS may not contain any Open Logical Volume Integrity Descriptors.

>>> [DVD] ERROR 5225 (ref. ECMA 3/10.1.7, 10.1.8, UDF 2.2.2.1) :
ERR_PVD_INTERCHANGE_LEVEL
Both the Interchange Level and the Maximum Interchange Level must be one. Only one PVD will exist.

>>> [DVD] ERROR 5226 (ref. ECMA 3/10.1.9, 10.1.10, 1/7.2.11, UDF 2.2.2.3 2.2.4) :
ERR_PVD_CSL
Both the Character Set List and the Maximum Character Set List must be one.

>>> [DVD] ERROR 5227 (ref. UDF 2.2.2.5 ECMA 3/10.1.11) :
ERR_PVD_VSI
The first 8 characters must form the CS0 hexadecimal representation of a 32-bit value.

This check is not yet implemented.

>>> [DVD] ERROR 5228 (ref. ECMA 3/10.1.20) :
ERR_PVD_PREDECESSOR
For every PVD, the Predecessor Volume Descriptor Sequence Location refers to the previous extent.

>>> [DVD] ERROR 5229 (ref. ECMA 3/10.1.21) :
ERR_PVD_FLAGS
The Flags of the PVD must be equal to one, only one PVD will exist.

>>> [DVD] ERROR 5231 (ref. UDF 2.2.4, 2.2.7.2.2 ECMA 3/10.4) :
ERR_IUVD_IDENTIFIER_NOT_CONSISTENT
The LogicalVolumeIdentifier of the Implementation Use Volume Descriptor with Implementation Identifier "*UDF LV Info" must be identical to the LogicalVolumeIdentifier of the Logical Volume Descriptor.

>>> [DVD] ERROR 5232 (ref. UDF 2.2.7.2.2 ECMA 3/10.4) :
ERR_IUVD_NOT_FOUND [ECMA] 3/10.4, [UDF] 2.2.7.2.2
An Implementation Use Volume Descriptor must exist with Implementation Identifier "*UDF LV Info".

>>> [DVD] ERROR 5235 (ref. DVD-2 2.6.5 ECMA 3/10.5.3, UDF 1.2) :
ERR_PD_CONTENTS
The Partition Contents must be equal to "+NSR02".

>>> [DVD] ERROR 5236 (ref. ECMA 3/10.5.7, UDF 2, DVD-2 2.6.4) :

ERR_PD_ACCESS_TYPE

The Access Type of a PD must be smaller than four.

>>> [DVD] ERROR 5237 (ref. DVD-2 2.1 9. 10. ECMA 3/10.5.6) :
ERR_PD_CONTENTS_USE [ECMA 3/10.5.6, DVD 2.1]

In case of DVD-2 ([DVD 2.1], items 9 and 10), the Unallocated Space Table, Unallocated Space Bitmap, Freed Space Table, or Freed Space Bitmap may not be recorded.

>>> [DVD] ERROR 5238 (ref. DVD-2 2.1 3.) :
ERR_PD_NUMBER

In case of DVD-2 ([DVD] 2.1, item 3), the Partition Number must be zero.

>>> [DVD] ERROR 5239 (ref. ECMA 3/10.5.3) :
ERR_PD_NO_VOLUME_SPACE_ALLOCATED

The Partition Flags must indicate that a volume space is allocated.

>>> [DVD] ERROR 5240 (ref. UDF 2.2.4.2 ECMA 3/10.6) :
ERR_LVD_BLOCK_SIZE

The Logical Block Size must be equal to the Logical Sector Size.

>>> [DVD] ERROR 5241 (ref. DVD-2 2.6.7, UDF 2, OR ECMA 3/10.6.8, 10.6.9. 10.7.3) :
ERR_LVD_MT_L_N_PM

According to [UDF], the Map Table Length shall not be less than the number of **Partition Maps** times six. Only type one Partition Maps are used, these are six bytes long.

According to [DVD], the Map Table Length is exactly six. For [DVD], only one partition exists.

>>> [DVD] ERROR 5245 (ref. DVD-2 2.6.8, CMA 3/10.7.3 UDF 2.2.4.6) :
ERR_PM_FIELDS

The Partition Map Type of a Logical Volume Descriptor must be one and the Partition Map Length must be six. In case if DVD-ROM, the Volume Sequence Number must be one.

>>> [DVD] ERROR 5246 (ref. ECMA 3/10.7.3.4) :
ERR_PM_MISMATCH

There must be a **Partition Descriptor** which has the same value for the **partition number**. Every partition map must be referred to by a **Partition Descriptor**.

This check is not yet implemented.

>>> [DVD] ERROR 5250 (ref. DVD-2 2.6.9, ECMA 3/10.8
ERR_USD_FREE_SPACE

The value of Number of Allocation Descriptors of an **Unallocated Space Entry** must be zero.

>>> [DVD] ERROR 5255 (ref. ECMA 10.10.6) :
ERR_LVID_NPARTITIONS_INCONSISTENT

The **Number of Partitions** of a **LVID** must be equal to the number of partitions in a **LVD**.

This check is not yet implemented.

>>> [DVD] ERROR 5256 (ref. ECMA 10.10.8, 10.10.9) :
ERR_LVID_SIZES_INCONSISTENT

The values **Free Space Table** and **Size Table** of the **LVID** must be consistent. These two sets of values need to be consistent with the partition descriptors.

This check is not yet implemented.

>>> [DVD] ERROR 5257 (ref. ECMA 3/10.10.8)
ERR_LVID_INCONSISTENT

The number of files and/or the number of directories is not consistent with the number of entries in the file set of part 4.

This check is not yet implemented.

9.3.21.1.4 DVD Filesystem ECMA4 checks

>>> [DVD] ERROR 5301 (ref. ECMA 4/7.1) :
ERR_LB_ADD_OUT_OF_RANGE
The **Partition Reference Number** of a lb_addr must be smaller than the number of partitions in the **LVID**. Furthermore, the **Logical Block Number** must be smaller than the number of logical blocks in the partition.

A mapping from partition to the number of blocks in that partition is maintained. This mapping is used to verify the validity of the address.

>>> [DVD] ERROR 5304 (ref. ECMA 4/8.3.1) :
ERR_FILE_SET_DESCRIPTORS_DIFFER
Any two prevailing instances of a **File Set Descriptor** may not specify the same file set identification.

>>> [DVD] ERROR 5305 (ref. ECMA 4/8.3.1) :
ERR_FILE_SET_NO_NUMBER_ZERO
One of the **File Set Descriptors** must have the value zero for its **File Set Number**.

>>> [DVD] ERROR 5306 (ref. ECMA 4/8.3.1) :
ERR_FILE_SET_DOUBLE_IDENTIFICATION
All File Set Descriptors with identical File Set Descriptor Numbers must be identical.

An event is generated when a sequence of FSD has been parsed. Checks like this one over more than one FSD can be performed.

>>> [DVD] ERROR 5307 (ref. UDF 2.3.2 ECMA 4/8.3.1
ERR_FILE_SET_DOMAIN_FLAGS
The **flags** of a **File Set Descriptor** must be set so it supports CSO character sets [UDF 2.1.2].

>>> [DVD] ERROR 5308 (ref. UDF 3.3 ECMA 4/8.3.1
ERR_FILE_SET_MULTIPLE_DESCRIPTORS
The number of File Set Descriptors in a File Set Descriptor Sequence must be precisely one.

>>> [DVD] ERROR 5309 (ref. DVD-2 3.3 ECMA 4/8.3.1
ERR_FILE_SET_NO_TERMINATOR
A File Set Descriptor Sequence must have a single Terminator Descriptor.

>>> [DVD] ERROR 5310 (ref. ECMA 4/14.1.10) :
ERR_FILE_SET_IDENTIFIER_NOT_CONSISTENT
The LogicalVolumeIdentifier of the File Set must be identical to the LogicalVolumeIdentifier of the Logical Volume Descriptor.

>>> [DVD] ERROR 5315 (ref. ECMA 4/8.6) :
ERR_FID_DOUBLE_IDENTIFIER
Every two File Identifier Descriptors must have a different File Identifier.

>>> [DVD] ERROR 5316 (ref. UDF 2.3.4.1 ECMA 4/8.6)
ERR_FID_FILE_VERSION_ NUMBER
The FileVersionNumber of a File Identifier Descriptor must be one.

>>> [DVD] ERROR 5317 (ref. ECMA 4/8.6) :
ERR_FID_DOUBLE_PARENT
The number of File Identifier Descriptors describing the parent must be one.

>>> [DVD] ERROR 5318 (ref. ECMA 4/8.6) :
ERR_FID_CSI
A File Entry specifying a file in which a directory is recorded shall not specify a Character Set Information Extended Attribute.

>>> [DVD] ERROR 5320 (ref. ECMA 4/8.7) :
ERR_FE_DATA_IN_DESCRIPTOR
The parser is not able to handle the data of a user file if it is recorded in the File Entry itself.

>>> [DVD] ERROR 5321 (ref. ECMA 4/8.10:
ERR_NUMBER_DE_INCORRECT
The number of ICB entries overflow the ICB extent.

>>> [DVD] ERROR 5325 (ref. UDF 2.3.2.1 ECMA 4/1.4.1) :
ERR_FSD_INTERCHANGE_LEVEL
Both the Interchange Level and the Maximum Interchange Level of a File Set Descriptor must be three.

>>> [DVD] ERROR 5326 (ref. UDF 2.3.2.3, ECMA 4/14.1) :
ERR_FSD_CSL
Both the Character Set List and the Maximum Character Set List of a File Set Descriptor must be one.

>>> [DVD] ERROR 5327 (ref.DVD-2 1.5.6, UDF 2.3.1.1, ECMA 3/7.2.5) :
ERR_FSD_TAG_SERIAL_NUMBER
The Tag Serial Number must be equal to zero.

>>> [DVD] ERROR 5328 (ref. DVD-2 3.3.1) :
ERR_FSD_FILE_SET_NUMBER
The File Set Number must be equal to zero.

>>> [DVD] ERROR 5332 (ref. ECMA 4/14.4) :
ERR_FID_L_IU
The length of the Implementation Use of a FID must be a multiple of four. Furthermore, it must be large enough to contain a regid.

>>> [DVD] ERROR 5335 (ref. ECMA 4/14.6) :
ERR_AED_PREVIOUS
Each Allocation Extent Descriptor in a sequence holds the location of its successor, except the head of the sequence.

>>> [DVD] ERROR 5341 (ref. UDF, DVD-2 A.8, UDF 2.4.5.1, ECMA 4/1.6.2) :
ERR_ICB_TAG_STRATEGY_TYPE
There are only two allowed strategies for an ICB: the value of Strategy Type must be either 4 or 4096.

>>> [DVD] ERROR 5342 (ref. UDF 6.6, DVD-2 A.8, ECMA 4/14.6.3) :
ERR_ICB_TAG_STRATEGY_PARAMETER
When the value of Strategy Type of an ICB tag is 4096, the value of the Strategy Parameter must be one.

>>> [DVD] ERROR 5343 (ref. UDF 6.6, ECMA 4/14.6.4) :
ERR_ICB_TAG_MAXIMUM_ENTRIES
When the value of Strategy Type of an ICB tag is 4096, the value of the Maximum Number of Entries must be two. Furthermore, there may not be more entries recorded in an ICB than the Maximum Number of Entries.

>>> [DVD] ERROR 5344 (ref. ECMA 4/14.6.8) :
ERR_ICB_TAG_FILE_TYPE
The value of File Type of an ICB tag must be smaller than 13.

>>> [DVD] ERROR 5345 (ref. ECMA 4/14.6.4) :
ERR_ICB_TAG_PARENT_ICB
Each member of a sequence of IBC's contains a reference to its predecessor (except the head of the sequence).

>>> [DVD] ERROR 5346 (ref. DVD-2 A.9, UDF 2.3.5.4, ECMA 4/14.6.4) :
ERR_ICB_TAG_FLAGS
The Flags, of the ICB tag, Sorted (Bit 4), Transformed (BIT 11), and Multi-versions (BIT12) must be equal to zero.

>>> [DVD] ERROR 5348 (ref. ECMA 4/14.9) :

ERR_FE_FILE_LINK_COUNT
The number of FID identifying the ICB which refers to the File Entry must be equal to the value of File Link Count of that File Entry.

>>> [DVD] ERROR 5349 (ref. UDF 2.3.7.0-3, ECMA 4/14.9.2) :
ERR_FE_RECORD
The values for RecordFormat, RecordDisplayAttributes, and RecordLength of a File Entry must be equal to zero.

>>> [DVD] ERROR 5350 (ref. ECMA 4/14.9.10) :
ERR_FE_INFORMATION_LENGTH
The value of the Information Length of the FE must be equal to the sum of the Information Lengths of the Allocation Descriptors.

>>> [DVD] ERROR 5351 (ref. ECMA 4/14.9.11, 12.1) :
ERR_FE_LOGICAL_BLOCKS
The value of Logical Blocks Recorded must be equal to the sum of the blocks in the recorded Allocation Descriptors.

>>> [DVD] ERROR 5352 (ref. ECMA 4/14.9.12) :
ERR_FE_DATE_AND_TIME
The Access, Modification, or Attribute Data and Time of the FE must be as least the File Creation Date and Time specified in the File Times Extended Attribute.

>>> [DVD] ERROR 5353 (ref. DVD-2 3.5.1, ECMA 4/14.9.15) :
ERR_FE_CHECKPOINT
The value of Checkpoint of a FE must be at least one.

>>> [DVD] ERROR 5354 (ref. ECMA 4/14.9.18) :
ERR_FE_ENTRIES_NOT_IDENTICAL
All FE with the same Unique Id must specify the same file.

>>> [DVD] ERROR 5355 (ref. UDF 3.3.3.4, ECMA 4/14.9.18) :
ERR_FE_UNIQUE_ID
The value of the Unique Id of a FE must be zero if the FE identifies the root directory. Furthermore, the value of the Unique Id may not be an element of {1..15}.

>>> [DVD] ERROR 5356 (ref. ECMA 4/14.9.19) :
ERR_FE_ATTRIBUTE_LENGTH
The value of Length of Extended Attributes must be an integral multiple of 4.

>>> [DVD] ERROR 5357 (ref. ECMA 4/14.9.22) :
ERR_FE_EXTENT_LOCATION
Any unrecorded or unallocated allocation descriptor must have its Extent Location set to zero.

>>> [DVD] ERROR 5358 (ref. DVD-2 3.5.1, A.7, ECMA 4/14.9.22) :
ERR_FE_NDESCRIPTORS
In case of [DVD], only short allocation descriptors may be used.

>>> [DVD] ERROR 5359 (ref. DVD-2 3.5.4, ECMA 4/14.9.22) :
ERR_FE_PERMISSIONS
The Permissions field of a FE must comply to the requirements presented in the table of [DVD 3.5.4].

>>> [DVD] ERROR 5360 (ref. ECMA 4/14.10.1) :
ERR_EA_LOCATION
The following requirements exist with respect to the attributes of the Extended Attribute Header descriptor:

- 24 <= Implementation Attributes Location
- Implementation Attributes Location <= Application Attributes Location
- Application Attributes Location <= Length of Extended Attributes (of the FE)

>>> [DVD] ERROR 5361 (ref. DVD-2 3.6, UDF 3.3.4, ECMA 4/14.10.1, UDF 3.3.4) :
ERR_EA_ATTRIBUTE_TYPE
In the first area of the extended attributes, attributes with values for the Attribute Type 1, 5, 6, and 12 are allowed. In the second area (Implementation Attributes), attributes with values for the Attribute Type within the range [2048..65535] are allowed. In the third area (Application Attributes), attributes with values for the Attribute Type starting at 65536 are allowed.

>>> [DVD] ERROR 5362 (ref. ECMA 4/14.10.1) :
ERR_EA_ATTRIBUTE_SUBTYPE
If the type of the extended attribute is part of {1, 3, 5, 6, 12, 2048, 65536}, the value of the Attribute Subtype must be one.

>>> [DVD] ERROR 5363 (ref. ECMA 4/14.10) :
ERR_EA_ATTRIBUTE_LENGTH
The value of Attribute Length of each attribute must correspond to the actual length of the attribute.

>>> [DVD] ERROR 5364 (ref. UDF 3.3.4.5.1) :
ERR_EA_HEADER_CHECKSUM
Header checksum error.

>>> [DVD] ERROR 5365 (ref. DVD-ROM 3.6) :
ERR_EA_ATTRIBUTE_NOT_FOUND
Extended Attribute ‘attribute string’ not found

>>> [DVD] ERROR 5366 (ref. DVD Table 3.6.4-2) :
ERR_EA_CGMS_INFORMATION
CMGS Information field is false.

>>> [DVD] ERROR 5367 (ref. DVD-2 3.6.4) :
ERR_EA_CGMS_TYPE
CMGS Data Structure Type should be ‘0’.

>>> [DVD] ERROR 5368 (ref. DVD-2 3.6.4) :
ERR_EA_CGMS_PROTECTION
CMGS Protection System Information is larger than ‘1’.

>>> [DVD] ERROR 5370 (ref. ECMA 4/14.14.1.2):
ERR_AD_OUT_OF_RANGE
When the value of Extent Length is equal to zero, the Extent Position must also be equal to zero. Analogous for [ECMA] 4/14.14.2.2.

>>> [DVD] ERROR 5370 (ref. ECMA 4/14.16.1.1):
ERR_AD_WRONG_TYPE
AD Type describes the wrong type.

>>> [DVD] ERROR 5375 (ref. ECMA 4/14.16.1.1) :
ERR_PATHNAME_TYPE
The value of Component Type of a Path Component must be larger than zero and smaller than six.

>>> [DVD] ERROR 5376 (ref. ECMA 4/14.16.1.2) :
ERR_PATHNAME_LENGTH
If the value of Component Type of the Path Component does not equal zero or five, the value of Length of Component Identifier must be zero. If the value of Component Type equals five, the value of Length of Component Identifier must be larger than zero.

>>> [DVD] ERROR 5377 (ref. ECMA 4/14.16.1.3) :
ERR_PATHNAME_VERSION
The value of Component File Version Number of a Path Component must be smaller than 32,768.

9.3.21.2 ISO 9660 File System Checks

>>> [DVD] ERROR 5501 :
ERR_ISO_RESERVED_NOT_NULL
Reserved bytes must be all NULL bytes. This also holds for unused and padding fields. Because this requirement occurs more than once in the standard, no specific reference is given.

>>> [DVD] ERROR 5502 (ref. ISO 6.1.2) :
ERR_ISO_LOGICAL_SECTOR_SIZE
The Logical Sector Size must be $2048*2^n$, where n is an integer which is at least zero ([ISO] 6.1.2).

This can not be verified. The descriptors which form the ISO file system do not contain information about the logical sector size.

>>> [DVD] ERROR 5503 (ref. ISO 6.2.2) :
ERR_ISO_LOGICAL_BLOCK_SIZE
The Logical Block Size must comply to two requirements:

1. The Logical Block Size must be $512*2^n$, where n an integer which is at least zero.
2. The Logical Block Size may not be larger than the logical sector size ([ISO] 6.2.2).

>>> [DVD] ERROR 5504 (ref. ISO 6.2.2) :
ERR_ISO_LBS_INCONSISTENT
The Logical Block Size is recorded in every PVD/SVD of the VRA. These recorded values must be the same ([ISO] 6.2.2).

>>> [DVD] ERROR 5505 (ref. ISO 6.7.1.1) :
ERR_ISO_NO_PVD
At least one Primary Volume Descriptor must be defined in the VRA ([ISO] 6.7.1.1).

>>> [DVD] ERROR 5506 (ref. ISO 6.7.1.5) :
ERR_ISO_NO_VDST
The sequence of descriptors in the VRA must be terminated by at least one Volume Descriptor Set Terminator ([ISO] 6.7.1.5).

>>> [DVD] ERROR 5507 (ref. ISO 6.8.1.3) :
ERR_ISO_DIR_LENGTH
The Data Length of a Directory Record must be a multiple of the Logical Block Size ([ISO] 6.8.1.3).

>>> [DVD] ERROR 5508 (ref. ISO 6.8.2.1) :
ERR_ISO_PATH_LENGTH
The total length of a File Identifier and the Directory Identifiers and the number of directories (this is the entire path of a file) may not exceed 255 characters. ([ISO] 6.8.2.1).

This is analogous verified as the possible error described in section **Error! Reference source not found.**.

>>> [DVD] ERROR 5509 (ref. ISO 6.8.2.1) :
ERR_ISO_DIR_LEVELS
The number of levels of a directory may not exceed eight ([ISO] 6.8.2.1).

>>> [DVD] ERROR 5510 (ref. ISO 6.8.2.2) :
ERR_ISO_DIR_DR
The Directory Records in a directory have to meet the following requirements (ISO 6.8.2.2):

1. At least two DR must be defined.
2. The first DR of the directory shall describe the directory itself and shall have a Directory Identifier consisting of a single ‘0’h byte.
3. The second DR of the directory shall describe the parent directory and shall have a Directory Identifier consisting of a single ‘1’h byte. If the root directory is evaluated, it shall also describe the directory itself.

>>> [DVD] ERROR 5511 (ref. ISO 7.6) :
ERR_ISO_DIRECTORY_IDENTIFIER
A Directory Identifier contains an error.

>>> [DVD] ERROR 5512 (ref. ISO 7.3) :
ERR_ISO_BOTH_BYTE_ORDER
When a 16 or 32 bit numerical value is recorded both LB and MB, the two recorded values must be equal ([ISO] 7.3).

>>> [DVD] ERROR 5513 (ref. ISO 7.3, Annex A) :
ERR_ISO_D_CHARACTERS
A sequence of d-characters must comply to the ISO standard 646 (this standard is part of [ISO], Annex A). The numerical value of each character must be a part of the collection: {‘30h’..’39h’, ‘41h’..’5Ah’, ‘5Fh’} ([ISO] 7.3).

>>> [DVD] ERROR 5514 (ref. ISO 7.3, Annex A) :
ERR_ISO_A_CHARACTERS
A sequence of a-characters must comply to the ISO standard 646 (this standard is part of [ISO], Annex A). The numerical value of each character must be a part of the collection: {‘20h’..’22h’, ‘25h’..’2Fh’, ‘30h’..’3Fh’, ‘41h’..’5Ah’, ‘5Fh’} ([ISO] 7.3).

>>> [DVD] ERROR 5515 (ref. ISO 8.4.20-8.4.22, 8.5.13-8.5.15) :
ERR_ISO_FILE_NOT_FOUND
When a PVD or SVD defines a file for the Publisher Identifier, Data Preparer Identifier, or the Application Identifier, this file must be located in the root directory to which the PVD/SVD refers ([ISO] 8.4.20-8.4.22 and 8.5.13-8.5.15).

>>> [DVD] ERROR 5516 (ref. ISO 8.1.1) :
ERR_ISO_DESCRIPTOR_TYPE
The value of Volume Descriptor Type must be part of the collection {0..3, 255} ([ISO] 8.1.1).

>>> [DVD] ERROR 5517 (ref. ISO 8.1.3) :
ERR_ISO_DESCRIPTOR_VERSION
The value of the Volume Descriptor Version must be 1 ([ISO] 8.1.3). Given all the possible descriptors of the [ISO] standard, the value of the VDV must be 1.

>>> [DVD] ERROR 5518 (ref. ISO 8.4.26.1 Table 5) :
ERR_ISO_DATE_TIME
The structure defined in table 5 of ([ISO] 8.4.26.1) must comply to:

- The string year must represent a value which is part of 1..9999.
- The string month must represent a value which is part of 1..12.
- The string day must represent a value which is part of 1..31.
- The string minute must represent a value which is part of 0..59.
- The string second must represent a value which is part of 0..59.
- The sting Hundredths of a second must represent a value which is part of 0..99.
- The string Offset must represent a value which is part of -48..52.

9.3.21.2.1 Boot Record

There is not much to verify about the Boot Record. The common VRA descriptor fields must be verified. Furthermore, the two attributes Boot System Identifier and Boot Identifier containing a-characters must be verified.

9.3.21.2.2 Primary Volume Descriptor

This sections lists the verification for a PVD.

>>> [DVD] ERROR 5519 (ref. DVD-2 A.13) :

ERR_ISO_SYSTEM_IDENTIFIER
The System Identifier of the PVD must be set to all '20'h bytes due to a DVD-Video requirement ([DVD-2] A.13).

>>> [DVD] ERROR 5521 (ref. ISO 8.4.20-8.4.22 and 8.5.13-8.5.15) :
ERR_ISO_FILE_NAME
If the first byte of the Publishers Identifier, Data Preparer Identifier, or Application Identifier is '5F'h, this field shall specify a file described with at most eight d-characters as a File Name and at most three d-characters as a File Name Extension ([ISO] 8.4.20-8.4.22 and 8.5.13-8.5.15).

>>> [DVD] ERROR 5522 (ref. ISO 7.6) :
ERR_ISO_FILE_IDENTIFIER
A File Identifier contains an error.

>>> [DVD] ERROR 5523 (ref. ISO 8.4.30) :
ERR_ISO_FSV
The value of the attribute File Structure Version must be 1 ([ISO] 8.4.30).

9.3.21.2.3 Supplementary Volume Descriptor

The requirements with respect to the SVD are analogous to the PVD. However, an additional requirement exists.

>>> [DVD] ERROR 5524 (ref. ISO 8.5.3) :
ERR_ISO_VOLUME_FLAGS
The bits one through seven of the Volume Flags must be zero ([ISO] 8.5.3).

9.3.21.2.4 Directory Record

This section describes the required verification with respect to the Directory Record.

>>> [DVD] ERROR 5525 (ref. DVD-2 A.14) :
ERR_ISO_DR_EARL
The Extended Attribute Record Length must be set to zero due to DVD-Video ([DVD-2] A.14).

>>> [DVD] ERROR 5526 (ref. ISO 9.1.5) :
ERR_ISO_DR_RDT
The following requirements have to be met due to [ISO] 9.1.5.

- The value of Month must be part of 1..12
- The value of Day must be part of 1..31
- The value of Hour must be part of 0..23
- The value of Minute must be part of 0..59
- The value of Second must be part of 0..59
- The value of Offset must be part of {-48, 52}

>>> [DVD] ERROR 5527 (ref. ISO 9.1.6) :
ERR_ISO_DR_FILE_FLAGS
If the DR describes a directory (bit position 1 equals zero), the flags for Associated File, Record, and Multi-Extent must be set to zero (bit positions 2, 3, and 7).

If no Extended Attribute Record is defined, the Record flag (bit position 3) and the Protection flag (bit position 4) must be set to zero. Due to [DVD-2], no extended attributes are allowed. Therefore, the Record (bit position 3) and Protection flag (bit position 4) must be set to zero.

The reserved flags (bit positions 5 and 6) must be set to zero.

These requirements are due to [ISO] 9.1.6.

>>> [DVD] ERROR 5528 (ref. DVD-2 A.14) :
ERR_ISO_DR_FUS
The File Unit Size must be set to zero due to DVD-Video ([DVD-2] A.14).

>>> [DVD] ERROR 5529 (ref. DVD-2 A.14) :
ERR_ISO_DR_IGS
The Interleave Gap Size must be set to zero due to DVD-Video ([DVD-2] A.14).

>>> [DVD] ERROR 5530 (ref. DVD-2 A.14) :
ERR_ISO_DR_SU
The System Use field must be six bytes long and must contain the Copyright Management Information ([DVD-2] A.14). Since the Copyright Management Information is six bytes, nothing else may be defined in the System Use field.

>>> [DVD] ERROR 5531 (ref. DVD-2 3.7.2) :
ERR_ISO_CMI_SUF
The Copyright Management Information of the system use field must meet the following requirements ([DVD-2] 3.7.2):

- The reserved bits of the CGMS Information field must be set to zero.
- The value of Data Structure Type of the CGMS must be set to zero.
- The Protection System Type of the Protection System Information must be either set to zero or one.
- The reserved bytes of the Protection System Information must be set to zero.

>>> [DVD] ERROR 5532 (ref. ISO 9.2) :
ERR_ISO_DIR_INCONSISTENT
The following attributes of each Directory Record must contain the same values if they refer to the same file ([ISO] 9.2):

- Existence, Directory, Associated, Record, and Reserved bits of the File Flags field.
- Padding Field

>>> [DVD] ERROR 5533 (ref. ISO 9.3) :
ERR_ISO_DIR_ORDER
The directories must be ordered according to the criteria listed below. The criteria with the lowest number have the highest priority ([ISO] 9.3).

1. Ascending with respect to the File Names.
2. Ascending with respect to the File Name Extensions.
3. Descending with respect to the File Version Numbers.
4. Descending with respect to the Associated File bit of the File Flags field.

9.3.21.2.5 Path Table Record

This section describes the verification which is required when a Path Table Record has been parsed. This verification does not include the consistency check of the VRA, Path Tables, and directories.

>>> [DVD] ERROR 5534 (ref. DVD-2 A.15) :
ERR_ISO_PTR_EARL
The Extended Attribute Record Length must be set to zero due to DVD-Video ([DVD-2] A.15).

>>> [DVD] ERROR 5535 (ref. ISO 8.4.13) :
ERR_ISO_PATH_TABLE_SIZE
The value of the Path Table Size must be equal to the sum of the sizes of all the Path Table Records in the path table ([ISO] 8.4.13).

>>> [DVD] ERROR 5536 (ref. ISO 6.8.2.2) :
ERR_ISO_DIR_LOCATION
The first DR must refer to the directory itself and the location of the directory, specified by this DR, must match the actual location of the directory ([ISO] 6.8.2.2).

>>> [DVD] ERROR 5537 (ref. ISO 6.8.2.2) :
ERR_ISO_DIR_PARENT
The second DR must refer to the parent and the by this DR specified location must match the actual location of the parent directory ([ISO] 6.8.2.2).

>>> [DVD] ERROR 5538 (ref. ISO 6.9) :
ERR_ISO_DR_PTR_INCONSISTENT
The PVDs and SVDs of the VRA contain references to the root directory, the type L Path Table, and the type M Path Table. These three items must be consistent. They must be consistent on the attributes Location of Extent, File Identifier, and the DR Number. Via the reference of the root directory an entire directory tree is parsed. The directories contained in this directory tree must all be listed in the path tables ([ISO] 6.9).

>>> [DVD] ERROR 5539 (ref. ISO 6.9) :
ERR_ISO_PTR_INCONSISTENT
When an entry in a path table describes a file, the entry must be consistent whith the entry in the other path table which describes the same file.

9.3.22 DVD Xchecks

9.3.22.1 Strategy for getting correct Cell data

For a number of cross checks, Cell data is required from the PGCI tables. When a VOB is parsed, there is no direct way to retrieve this data, as Cells in a VOB do not map directly onto Cells in a PGC. The method for accessing the Cell data is described here.

The DSI.DSI_GI has the VOBU_VOB_IDN and VOBU_C_IDN encoded. These values identify the Cell in a VOB. The PGCI.C_POSIT contains the mapping from VOBU_VOB_IDN and VOBU_C_IDN to the Cell number in a PGC, which means that all PGCIs from the VTSI must be scanned. This results in the PGC number and Cell number. The Cell data can be found in the PGCI.C_PBIT.

The way to acces this data, using the xcheck methods is:

```
xcheck->C_IDN2Cell_number( xcheck, VOBU_C_IDN, VOBU_VOB_IDN);
```

This results in a `Cell_list` entry, that contains all information about the Cell from the PGCI data. From this `Cell_list` entry, all Cell attributes can be retrieved with the available xcheck methods.

9.3.22.2 General Cross Checks

>>> [DVD] SYSTEM ERROR 5601 (ref. DVD_xcheck) :
ERR_DVD_XCHECK_DISABLED
The cross check file could not be read! Make sure the 'dvd_verif_xdata.info' exists. This error is also reported when the file does exists, but the following cases occur:

- The requested data was not present in the file, eg. VTSI or VMGI data.
- The requested data is present, but is in a wrong format. This is probably due to an edited cross check file, or a version of the cross check file generated with an other version of the verifier.

This will be reported as a system error.

>>> [DVD] INFORMATION 5602 (ref. DVD_xcheck) :
ERR_DVD_XCHECK2_DISABLED
An error occurred while trying to write to the cross check file. Cross checks will be disabled. This error is reported when the verifier can not create the cross check file. This can be due to:

- The verifier does not have write permission in the directory where the verifier was started, which will happen when verifying directly from a read-only device (e.g. DVD-ROM drive).
- Not enough space on the device was available to the verifier for the cross check file.
- Miscellaneous resons why the file cound not be written, such as malfunctioning device, device not found, etc.

This will be reported as a system error.

>>> [DVD] ERROR 5603 (ref. DVD-3 4.1.5.3) :
ERR_DVD_XCHECK_ILLEGAL
Cross Check for a field failed during comparison between values for identical fields in different files. This error is reported when verifying VMGI.VTS_ATRT values against the same data in the VTSI.VTSI_MAT BP 34-37 and BP 256-1023. All values should be equal.

>>> [DVD] ERROR 5604 (ref. DVD_xcheck) :
ERR_DVD_SCRIPT_XCHECK_CONFLICT
The parameters in the script-file should be equal to the corresponding fields from the cross check file. When an inconsistency is found, this is reported and the value from the script-file is used. This gives the user the possibility to override the cross check values in case these were found to be incorrect, or just for testing purposes. This is reported as an information message.

>>> [DVD] ERROR 5605 (ref. N/A) :
ERR_DVD_XCHECK_NO_CELL_REF
The current Cell is not referenced by a PGC in the VTS for the current domain and therefore cannot be played

The message reports that a VOB is present in the VOBS, but not used in the definition of the PGCs in the VTSI. This means that you cannot access the data in that VOB from the navigation data point of view. The data is not usable and therefore not really usefull and thus a waste of diskspace. This message will be reported as an ODDITY.

9.3.22.3 VTSI Cross Checks

>>> [DVD] ERROR 5609 (ref. DVD_xcheck) :
ERR_DVD_XCHECK_ILL_PGCN
The PGCN is not specified in the VTSI, in the current domain.

>>> [DVD] ERROR 5610 (ref. DVD-3 4.2.5) :
ERR_DVD_XCHECK_TMAPT_ABSENT
No VTS_TMAPT found in VTSI, but it is mandatory if the VMGI.TT_SRP.TT_TY for the current title, equals 0, indicating a sequential PGC title.

>>> [DVD] ERROR 5611 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_VTSTTN_LARGE
VTS_TTN from VMGI is larger than the number of VTS_TTU_SRP from the VTSI.

>>> [DVD] ERROR 5612 (ref. DVD_xcheck) :
ERR_DVD_XCHECK_VTSN_LARGE
The current VTS number is not specified in the VMGI. All VTS's should be specified in the VMGI.

>>> [DVD] ERROR 5613 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_VTSN_ILL
VTS_TTN from the current VTS is not found in the VMGI.

>>> [DVD] ERROR 5614 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_PTTN_ILL
PTT_Ns from VMGI does not equal to number of PTT_SRP in the current Title from the current VTS.

>>> [DVD] ERROR 5615 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_PGCN_ILL
The PGCN from PTT_SRP must be assigned consequtively, starting from '1', when the Title_Type describes a One_Sequential_PGC_Title.

>>> [DVD] ERROR 5616 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_PGN_ILL
The PGN should be assigned consecutively for each PGCN, starting from '1', when the Title_Type describes a One_Sequential_PGC_Title.

>>> [DVD] ERROR 5617 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_PGN_NOT_ONE
The PGN for a new PGCN should be '1', when the Title_Type describes a One_Sequential_PGC_Title.

>>> [DVD] ERROR 5618 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_RANDOMPGCN_ILL
The PGCN may only specify one PGN, thus the PGCN must be '1' larger than the previous PGCN, when the Title_Type describes a One_Random_PGC_Title, which allows only 1 PGC entry per PTT.

>>> [DVD] ERROR 5619 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_RANDOMPGN_ILL
The PGCN may only specify one PGN, thus the PGN must always be '1', when the Title_Type describes a One_Random_PGC_Title, which allows only 1 PGC entry per PTT.

>>> [DVD] ERROR 5620 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_BLOCKPGCN_ILL
The PGCN from the PTT_SRP should specify the first PGC from a block.

>>> [DVD] ERROR 5625 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_PGN_LARGE
The PGN from the PTT_SRP is larger than the number of Programs specified in PGC 'PGC number' of the 'Domain string'.

>>> [DVD] ERROR 5626 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_PTTN_NOT_FOUND
The current Cell (C_IDN='cell id', VOB_IDN='VOB id) belongs to PG#'program number' of PGC#'PGC number', but this PG is not referred to by a PTT.

All Programs should be refered to by a PTT.

>>> [DVD] ERROR 5627 (ref. DVD-3 4.2.2) :
ERR_DVD_XCHECK_TT_NOT_FOUND
The current Cell (C_IDN='cell id', VOB_IDN='VOB id') belongs to PG#'program number' of PGC#'PGC number', refers to a PTT that is found in TTU#'Title unit number', which is not specified in the VMGI.

9.3.22.4 Navigation Commands Cross Checks

>>> [DVD] ERROR 5651 (ref. DVD-3 4.6.4.1) :
ERR_DVD_XCHECK_NAVCMD_NS_BIG
The Navigation command GoTo or SetTmpPML specified a Navigation command number larger than the number of Navigation commands specified in the current PGC.

>>> [DVD] ERROR 5652 (ref. DVD-3 4.6.4.2) :
ERR_DVD_XCHECK_ILL_DOMAIN
The Navigation command specified a destination which could not be found in the current Domain. This error is reported for these Navigation commands:

- LinkPGCN, when the specified PGC is not present in the VTS.
- LinkPGN, when the specified PG is not present in the VTS.

>>> [DVD] ERROR 5653 (ref. DVD-3 4.6.4.2/3) :
ERR_DVD_XCHECK_ILL_BLOCK_MODE
The Navigation command is part of a block, but the Block_mode of the command is not correctly set. This error is reported when:

- LinkPGCN specifies a PGC, but this PGC's Block_mode should be '01b' (the first PGC in the block).
- LinkCN specifies a Cell, but this Cell's Block_mode should be '01b' (the first Cell in the block).
- CallSS specifies a Cell for resume, but this Cell's Block_mode should be '01b' (the first Cell in the block).

>>> [DVD] ERROR 5654 (ref. DVD-3 4.6.4.2/3) :
ERR_DVD_XCHECK_MAX_PTTN
The number of the PTT specified in the Navigation command is not legal for the current Title. The PTT values should be <99 in a sequential Title, or <999 in a random or shuffle Title. This error is reported for the Navigation commands:

- LinkPTTN.
- JumpVTS_PTT.

>>> [DVD] ERROR 5655 (ref. DVD-3 4.6.4) :
ERR_DVD_XCHECK_ENTRY_NOTFOUND
The Navigation command specified a value, that could not be found in the VTS. This error is reported when:

- The LinkPTTN Navigation command specified a non-present VTS number.
- The LinkPTTN Navigation command specified a non-present PTT number.
- The LinkPGN Navigation command specified a non-present Program number.
- The LinkCN Navigation command specified a non-present Cell number.
- The JumpTT Navigation command specified a non-present Title number.
- The JumpVTS_TT Navigation command specified a non-present VTS_TT number.
- The JumpVTS_PTT Navigation command specified a non-present Title number.
- The JumpVTS_PTT Navigation command specified a non-present PTT number.
- The JumpSS Navigation command specified a non-present PGC number, in case of a jump to the VMGM Domain.
- The JumpSS Navigation command specified a non-present Title number, in case of a jump to the VTSM Domain.
- The CallSS Navigation command specified a non-present Title number, in case of a call to the VTSM Domain.
- The CallSS Navigation command specified a non-present PGC number, in case of a call to the VMGM Domain.
- The SetNVTMR Navigation command specified a non-present PGC number.

The LinkPTTN Navigation command specified a non-present PTT number

>>> [DVD] ERROR 5656 (ref. DVD-3 4.6.4.2 (e)) :
ERR_DVD_XCHECK_ONE_ENTRY
This error reports a Navigation command that is not able to be performed, because only one entry to which the Navigation command refers exists. This error is reported when:

- The Link_S_LinkNextC Navigation command specifies to link the next Cell, but only 1 Cell exists in the PGC.
- The Link_S_LinkNextPG Navigation command specifies to link the next program, but only 1 program exists in the PGC.

>>> [DVD] ERROR 5657 (ref. DVD-3 4.6.4.2 (e)) :
ERR_DVD_XCHECK_NOT_EXISTS
This error reports a Navigation command that is not able to be performed, because the entry to which the Navigation command refers does not exist. This error is reported when:

- The Link_S_LinkNextPGC Navigation command specifies to link the next Program Chain, but the NextPGC field from the current PGC (cf. [DVD-3] 4.3.2 (6)) equals '0', meaning no next PGC.
- The Link_S_LinkPrevPGC Navigation command specifies to link the previous Program Chain, but the PrevPGC field from the current PGC (cf. [DVD-3] 4.3.2 (6)) equals '0', meaning no previous PGC.

>>> [DVD] ERROR 5658 (ref. DVD-3 4.6.4.2 (e)) :
ERR_DVD_XCHECK_NO_GOUP_PGC
This error reports the Link_S_LinkGoUpPGC Navigation command is not able to be performed, because the GoUpPGC to which the Navigation command refers is illegal. This error is reported when the Navigation

command specifies to link the GoUp Program Chain, but the GoUpPGC field from the current PGC (cf. [DVD-3] 4.3.2 (6)) equals '0', meaning no GoUp PGC.

>>> [DVD] ERROR 5659 (ref. DVD-3 4.6.4.2 (e)) :
ERR_DVD_XCHECK_GOUP_PGC_ILL
This error reports the Link_S_LinkGoUpPGC Navigation command is not able to be performed, because the GoUpPGC to which the Navigation command refers does not exist. This error is reported when the Navigation command specifies to link the GoUp Program Chain, but the PGC with the PGCN from the GoUpPGC field (cf. [DVD-3] 4.3.2 (6)) is not present in the current VTS.

>>> [DVD] ERROR 5660 (ref. DVD-3 4.6.4.3) :
ERR_DVD_XCHECK_NO_FP_PGCI
This error reports a Navigation command that is not able to be performed, because first play PGC does not exist. This error is reported when:

- The JumpSS Navigation command specifies a jump to the first play PGC (indicated by the Domain ID from the navigation command argument (cf. [DVD-3] 4.6.4.3 (e))), but no first play PGC exists in the current VTS.
- The CallSS Navigation command specifies a call to the first play PGC (indicated by the Domain ID from the navigation command argument (cf. [DVD-3} 4.6.4.3 (f))), but no first play PGC exists in the current VTS.

>>> [DVD] ERROR 5670 (ref. DVD-3 4.6.4.3) :
ERR_DVD_XCHECK_ENTRYPGC_NOTFOUND
This error reports a Navigation command that is not able to be performed, because the PGC to which the Navigation command refers is defined as an entry PGC. Only an entry PGC can be specified. This error is reported when:

- The JumpSS Navigation command specifies a Domain ID '00b' or '11b' (cf. [DVD-3] 4.6.4.3 (e)).
- The CallSS Navigation command specifies a Domain ID '00b' or '11b' (cf. [DVD-3] 4.6.4.3 (f)).

>>> [DVD] ERROR 5671 (ref. DVD-3 4.6.4.3) :
ERR_DVD_XCHECK_MENU_NOTFOUND
This error reports a Navigation command that is not able to be performed, because the Menu to which the Navigation command refers is not present in the current PGC. This error is reported when:

- The JumpSS Navigation command specifies a Domain ID '10b' (cf. [DVD-3] 4.6.4.3 (e)).
- The CallSS Navigation command specifies a Domain ID '01b' or '10b' (cf. [DVD-3] 4.6.4.3 (f)).

>>> [DVD] ERROR 5672 (ref. DVD-3 4.6.4.3 (f)) :
ERR_DVD_XCHECK_RESUME_ILL
This error reports the CallSS Navigation command is not able to be performed, because the Cell number for resume specified in the Navigation command is not present in the current PGC.

>>> [DVD] ERROR 5673 (ref. DVD-3 4.6.4.3 (f)) :

9.3.22.5 Audio Cross Checks

>>> [DVD] ERROR 5674 (ref. DVD-3 4.6.4.3) :
ERR_DVD_XCHECK_ENTRY_LARGE
The navigation command SetSYS_STN specified a value for one of the following fields that was larger than specified in the VMGI:

- ASTN, the number of the Audio stream.
- SPSTN, the number of the Sub-picture stream.
- AGLN, the number of Angles.

>>> [DVD] ERROR 5701 (ref. DVD-3 4.3.2-2) :
ERR_DVD_XCHECK_DEC_ASTN_NOT_FOUND
The Audio stream specified in the PGC (PGC_AST_CTLT) was not found in this VOB.

At the end of the VOBS, all audio streams that have the Availability_flag in the PGC_AST_CTL of the audio stream set to '1' will be checked against the audio streams that were present in the VOB. When audio streams should be available, but were not found, this error is generated.

>>> [DVD] ERROR 5702 (ref. DVD-3 4.3.2-2) :
ERR_DVD_XCHECK_ASTN_NOT_FOUND
An Audio stream was found in the VOB, but was not specified in the PGC (PGC_AST_CTLT).

When an audio frame header is encountered in the VOBS, the Availability_flag for that audio stream should be set to '1' in the PGC_AST_CTL of the audio stream. This error is generated when the Availability_flag is set to '0'.

>>> [DVD] ERROR 5703 (ref. DVD-3 4.1.1 BP 260 / 4.2.1 BP 260/516) :
ERR_DVD_XCHECK_RANDOM_ILL
This Navigation command is not allowed when the Title is a random PGC title. This error is reported for the CallSS Navigation command.

ERR_DVD_XCHECK_CHANNELS_ILL
The number of Audio_channels specified does not correspond with the number of audio_channels found in the VOB. This error is checked at:

- The end of the audio frame header, in case of a mono or stereo MPEG audio stream.
- The end of the multi channel extension header, in case of a multi channel MPEG audio stream.
- The end of the seven channel augmentation data, in case of a 7.1 MPEG-2 audio stream.
- The end of the Private-1 header, in case of Dolby AC-3 and LPCM streams.

The numbers reported in the error message are the real numbers of audio streams, not the value of the number_of_audio_channels field (which is normally 1 less then the actual number of audio channels).

>>> [DVD] ERROR 5704 (ref. DVD-3 4.1.1 BP 260 / 4.2.1 BP 260/516) :
ERR_DVD_XCHECK_DRC_ILL
The Audio attributes specified Quantization_DRC does not correspond with data found in the VOB. This error is generated in these cases:

- The Quantization_DRC equals '1', meaning dynamic range control bits available, but the audio stream does not contain these dynamic range control bits.
- The Quantization_DRC equals '0', meaning dynamic range control bits available, but the audio stream contains these dynamic range control bits.

9.3.22.6 Sub-picture Cross Checks

>>> [DVD] ERROR 5726 (ref. DVD-3 4.3.2-3) :

ERR_DVD_XCHECK_DEC_SPSTN_NOT_FOUND
The Sub-picture stream specified in the PGC (PGC_SPST_CTLT) was not found in this VOB.

At the end of the VOBS, all Sub-picture streams that have the Availability_flag in the PGC_SPST_CTL of the Sub-picture stream set to '1' will be checked against the Sub-picture streams that were present in the VOB. When Sub-picture streams should be available, but were not found, this error is generated.

>>> [DVD] ERROR 5727 (ref. DVD-3 4.3.2-3) :
ERR_DVD_XCHECK_SPSTN_NOT_FOUND
An Sub-picture stream was found in the VOB, but was not specified in the PGC (PGC_SPST_CTLT).

When an Sub-picture packet is encountered in the VOBS, the Availability_flag for that Sub-picture stream should be set to '1' in the PGC_SPST_CTL of the Sub-picture stream. This error is generated when the Availability_flag is set to '0'.

9.3.22.7 VOB Cross Checks

>>> [DVD] ERROR 5751 (ref. DVD-3 4.3.2) :
ERR_DVD_XCHECK_VOBU_SA_NOT_FOUND
The Start Address of the current VOB or VOBU does not correspond with any VOBU-Start Address specified in the VOBU_ADMAP table. Since each VOB starts with a VOBU, the start address of a VOB must be equal to one of the VOBU start addresses in the VOBU_ADMAP for the current VTS.

>>> [DVD] ERROR 5776 (ref. DVD-3 4.3.2-1 (1)) :
ERR_DVD_XCHECK_PGCI_NO_VOB
In a PGC without any VOB, a number of fields should be zero. This error is reported in these cases:

- The Number of Programs value does not equal '0'.
- The Number of Cells value does not equal '0'.
- The PG_playback_mode from the PGC_NV_CTL does not equal '0'.
- The Still_time_value from the PGC_NV_CTL does not equal '0'.

>>> [DVD] ERROR 5777 (ref. DVD-3 4.3.2-1 (2)) :
ERR_DVD_XCHECK_TCFLAG_ILL
The tc_flag does not correspond with the TV_system. The tc_flag indicates the number of frames per second, which is only valid for the correct TV_system. This error is reported when:

- The tc_flag equals 3, but the TV_system for the current VOBU is PAL.
- The tc_flag equals 1, but the TV_system for the current VOBU is NTSC.

The TV_system is specified in the VTSI field VTSM_V_ATR for VOBUs in a VTS menu VOB and the VTSI field VTS_V_ATR for VOBUs in a VTS title VOB.

>>> [DVD] ERROR 5778 (ref. DVD-3 4.3.2-2/3) :
ERR_DVD_XCHECK_AVAILABLE_STREAMS_ILL
The number of Available Audio or Sub-picture streams is larger than the number specified, which is found in the VTSI:

- VTSM_AST_Ns for Audio streams in a VTS menu VOB.
- VTS_AST_Ns for Audio streams in a VTS title VOB.
- VTSM_SPST_Ns for SP streams in a VTS menu VOB.
- VTS_SPST_Ns for SP streams in a VTS title VOB.

This error reports that more Audio or Sub-picture streams were found in the VOB, than specified in the VTSI, where the number of streams is specified. This number gives the number of possible streams, not the actual number. The number of Audio or Sub-picture streams in the VOB may threrefore be less than the number specified in the VTSI.

>>> [DVD] ERROR 5779 (ref. DVD-3 4.3.2-3) :
ERR_DVD_XCHECK_DEC_NR_ILL
The decoding sub picture stream number is illegal with the current Aspect_ratio. This error is reported if a field of the PGC_SPST_CTL should be '0'.

When the Aspect_ratio for the current VOB is 4:3, then these fields should be '0':

- Decoding_sub_picture_stream_for_Wide.
- Decoding_sub_picture_stream_for_Letterbox.
- Decoding_sub_picture_stream_for_Pan-scan.

When the Aspect_ratio for the current VOB is 16:9, then this fields should be '0':

- Decoding_sub_picture_stream_for_4:3.

The Aspect_ratio is specified in the VTSI field VTSM_V_ATR for VOBUs in a VTS menu VOB and the VTSI field VTS_V_ATR for VOBUs in a VTS title VOB.

>>> [DVD] ERROR 5780 (ref. DVD-3 4.3.2-1 (6)) :
ERR_DVD_XCHECK_PGC_NV_CTL_ILL
Fields from the PGC_NV_CTL should be '0' in a One_Sequential_PGC_Title. This error is reported when:

- The Next_PGC_number does not equal '0' in a One_Sequential_PGC_Title.
- The Previous_PGC_number does not equal '0' in a One_Sequential_PGC_Title.
- The GoUp_PGC_number does not equal '0' in a One_Sequential_PGC_Title.

To determine if a title is a One_Sequential_PGC_Title, the field TT_TY from the structure TT_PB_TY from the VMGI for the current Title should be '0'.

9.3.22.8 TMAP Cross Checks

>>> [DVD] ERROR 5801 (ref. DVD-3 4.2.5-3) :
ERR_DVD_XCHECK_TMAP_ENTRY_ILL
The MAP_ENA should describe the start address of the VOBU, where the presentation time corresponding to the MAP_EN is included, with RLBN from the first LB of the VTSTT_VOBS in the VTS.

Determination of the correctness of the MAP_ENA is done by calculating the time elapsed since the start of the PGC:

Total_Previous_Cell_time - cell_start_time + VOBU_E_PTM (ticks)

The value is compared to the calculated elapsed time for the current MAP_EN:

MAP_EN * TMU * 90000 (ticks)

When the elapsed time in the PGC exceeds the elapsed time for the MAP_EN, the start address of the current VOBU is compared to the MAP_ENA value, which should be identical. Also, the verifier is set to check the next MAP_EN.

>>> [DVD] ERROR 5802 (ref. DVD-3 4.2.5-3 (3)) :
ERR_DVD_XCHECK_TMAP_1ST_ANGLE_ONLY
The MAP_ENT should only describe MAP_ENAs for the first angle, skipping MAP_ENAs until current VOBU start address.

This error reports that the TMAP table in the VTSI specifies MAP_ENs for each angle in the angle block. According the [DVD] 4.2.5-3 (3) spec, only MAP_ENAs are allowed for angle 1. The remaining MAP_ENAs are skipped. This error is detected when the a MAP_ENA is smaller than the previous MAP_ENA.

>>> [DVD] ERROR 5803 (ref. DVD-3 4.2.5-3 (3)) :
ERR_DVD_XCHECK_TMAP_SKIPPED
The MAP_ENA cannot be used, because the stream is already past this position (current VOBU start address). Skipping MAP_ENAs until current VOBU start address.

This error could also be a result of 5802, but can also indicate a problem in the TMAP table. A MAP_ENA was specified that didn't belong in the table, because the current VOBU address should be used, according the calculations.

9.3.22.9 Cell Attribute Cross Checks

>>> [DVD] ERROR 5826 (ref. DVD-3 4.3.5-1) :
ERR_DVD_XCHECK_CELL_ADDRESS
A field specifying an address from the PGCI.C_PBI does not correspond with data from the VOB. This error is reported when:

- The C_FILVU_EA does not correspond with the end address of the first ILVU in the Cell.
- The C_LVOBU_SA does not correspond with the start address of the last VOBU in the Cell.
- The C_LVOBU_EA does not correspond with the end address of the last VOBU in the Cell.

These errors are checked at the end of each Cell. The C_FVOBU_SA is not checked here, because this address equals the Cell start address and is checked in **Error! Reference source not found.**

>>> [DVD] ERROR 5827 (ref. DVD-3 4.3.5) :
ERR_DVD_XCHECK_CELL_SA_NOT_FOUND
The Start Address of the current Cell does not correspond with the Start Address of the first VOBU in the Cell. This is checked at the start of a Cell. The start address of the Cell should be equal to the C_F_VOBU_SA in the PGCI.C_PBI, because each Cell starts with a VOBU.

>>> [DVD] ERROR 5828 (ref. DVD-3 4.3.5-1 (1)) :
ERR_DVD_XCHECK_CELL_BLOCK_MODE
The Cell_Block_mode of the Cell data in the PGCI does not correspond to the block mode of the current Cell. This error is reported when:

- The Cell_Block_mode equals '00b' (not a cell in a block), but the current Cell is found in an angle block. This is checked at the start of each Cell.
- The Cell_Block_mode does not equal '01b' (the first cell in a block) and the current Cell is found as the first Cell in an angle block. This is checked at the start of each Cell.
- The Cell_Block_mode does not equal '10b' (a cell in a block), but the current Cell is found in an angle block and is not the first or the last Cell in the angle block. This is checked at the start of each Cell.
- The Cell_Block_mode does not equal '11b' (the last cell in a block), but the previous Cell was found to be the last Cell in an angle block. This is checked at the end of each the interleaved block.

The Cell_Block_mode is specified in the PGCI.C_PBIT.CPBI.C_CAT.

>>> [DVD] ERROR 5829 (ref. DVD-3 4.3.5-1 (1)) :
ERR_DVD_XCHECK_CELL_BLOCK_TYPE
The Cell_Block_type of the Cell data in the PGCI does not correspond to the block type of the current Cell. This error is reported when:

- The Cell_Block_type equals '00b' (not part of a block), but the current Cell is found in an angle block.
- The Cell_Block_type equals '01b' (part of an angle block), but the current Cell is not found in an angle block.

These errors are checked at the start of each Cell. The Cell_Block_type is specified in the PGCI.C_PBIT.CPBI.C_CAT.

>>> [DVD] ERROR 5830 (ref. DVD-3 4.3.5-1 (1)) :
ERR_DVD_XCHECK_CELL_INTERLEAVED
The Interleaved_allocation_flag from the PGCI.C_PBIT.CPBI.C_CAT should correspond to the ILVU_flag from the DSI.SML_PBI.VOBU_SMLCAT

>>> [DVD] ERROR 5831 (ref. DVD-3 4.3.5-1 (1)) :
ERR_DVD_XCHECK_CELL_SEAMLS
The Seamless_playback_flag and the STC_discontinuity_flag specify an illegal value, for the current Cell. This error is reported when the Seamless_playback_flag and the STC_discontinuity_flag do not comply to this table:

Previous Cell	***Current Cell***	***Seamless playback flag***	***STC discontinuity flag***
Cell in angle block	Single	1	1
Single Cell	Cell in angle block	1	1
No Cell		0	1
Cell in angle block	Cell in angle block	1	0

This is checked at the start of each Cell.

>>> [DVD] ERROR 5832 (ref. DVD-3 4.3.5-1 (1)) :

ERR_DVD_XCHECK_CELL_STILL

Cell_still_time in previous cell should be '0' (no still), because it is the last cell of the program and the still time of the PGC of that program is not '0'.

>>> [DVD] ERROR 5833 (ref. DVD-3 4.3.5-1 (1)) :
ERR_DVD_XCHECK_CELL_CMDNUM

Cell_command_number is not present in any cell command of PGC. This error is reported when the Cell_command_number from the Cell is larger then the number specified in the PGC.PGC_CMDTI.C_CMD_Ns (cf. [DVD-3] 4.3.3-1).

>>> [DVD] ERROR 5834 (ref. DVD-3 4.3.5-1 (2)) :
ERR_DVD_XCHECK_CELL_PBTM_ILL

The C_PBTM of the current Cell should be equal to the C_ELTM of the last VOBU of the Cell increased with the duration of that VOBU.

9.3.22.10 GOP Cross Checks

>>> [DVD] ERROR 5876 (ref. DVD-3 4.2.5) :
ERR_DVD_XCHECK_FRAMERATE_ILL

The specified frame_rate does not correspond with the TV_system for this VOB. This error is reported when:

- The frame_rate equals '00b' and the TV_system equals '00b' (PAL).
- The frame_rate equals '10b' and the TV_system equals '10b' (NTSC).

>>> [DVD] ERROR 5877 (ref. DVD-3 4.2.5-3) :
ERR_DVD_XCHECK_LINE21DATA_ILL

Line21 user_data available in GOP for a field, while the corresponding line21_switch was not set in the VTSI. This error is reported when:

- Line21 user_data is available in the GOP for a top field, but the line21_switch_1 equals '0'.
- Line21 user_data is available in the GOP for a bottom field, but the line21_switch_2 equals '0'.

The line21_switch_1 and line21_switch_2 can be found in the VTSI_MAT.VTSM_V_ATR for a menu VOB and VTSI_MAT.VTS_V_ATR for a title VOB (cf. [DVD-3] 4.2.1-1).

9.3.22.11 Angle Cross Checks

>>> [DVD] ERROR 5901 (ref. DVD-3 4.2.5-3) :
ERR_DVD_XCHECK_ANGLE_ILL

The Number of Angles in the current ILVB does not correspond to the AGL_Ns in the title. The AGL_Ns is found in the VMGI.TT_SRPT.TT_SRP (cf. [DVD-3] 4.1.2-2) with the current VTSN and VTS_TTN.

9.3.22.12 File System Cross Checks

Both file systems, which are placed on a DVD disc, describe a tree of directories and files. The entire path and identifier of a file is used as a key. Both file systems describe attributes of files. These attributes can be compared. When these descriptions are not consistent, an error message must be generated. To perform this verification, the xcheck structure is used, see [xcheck].

The following strategy is used to perform this verification. The verification can be divided into the following stages.

1. When a File Identifier and a File Entry (ECMA/UDF file system) have been parsed, two events are generated by the parser. First, the event EVT_FID_FE_PATH is generated. This event is accompanied by a reference to entire path of the file described by the FID and FE for which an event is generated directly after this event. The path is stored in the cross check structure and can be used in the future. Second, the event EVT_FID_FE is generated when a FID and its corresponding FE have been parsed. Using these three structures (the path name, the FID and the FE) a structure is created to store this information about a single file in the cross check object.

2. When a Directory Record (ISO file system) has been parsed, the event EVT_DR_PATH is generated. This event is accompanied by a DR and the path of the file which is described by the DR. This information is also stored in the xcheck structure.
3. When the entire file system has been parsed, the event EVT_FS_PARSED is generated. The information gathered from parsing both file systems is stored in the xcheck file.
4. When the cross check needs to be performed, the information about the files (found in both file systems) is restored from file and the verification is performed. The entire path of a file is used as a key. For every file in either one of the file systems the corresponding file information is examined. If there is no corresponding description in the other file system, an error is generated. Furthermore, when the corresponding file information is not consistent, an error message is generated.

The error messages which can be detected when the cross checks are performed, are presented in this section.

>>> [DVD] ODDITY 5951:
ERR_FS_XCHECK_LOCATION
When both file systems describe the same file, the locations of the file described by both file systems must be the equal.

>>> [DVD] ODDITY 5952:
ERR_FS_XCHECK_SIZE
When both file systems describe the same file, the size of the file described by both file system must be the equal.

>>> [DVD] ODDITY 5953:
ERR_FS_XCHECK_FVN
When both file systems describe the same file, the value of the attribute File Version Number described by both file system must be equal.

>>> [DVD] ODDITY 5954:
ERR_FS_XCHECK_DIRECTORY_FLAG
When both file systems describe the same file, the value of the attribute Directory Flag described by both file systems must be equal.

>>> [DVD] ODDITY 5955:
ERR_FS_XCHECK_EXISTENCE_FLAG
When both file systems describe the same file, the value of the attribute Existence Flag described by both file systems must be equal.

>>> [DVD] ODDITY 5956:
ERR_FS_XCHECK_CGMS
When both file systems describe the same file, the value of the attribute Copyright Generation Management System described by both file systems must be equal.

>>> [DVD] ODDITY 5957:
ERR_FS_XCHECK_CM
When both file systems describe the same file, the value of the attribute Copyrighted Material described by both file systems must be equal.

>>> [DVD] ODDITY 5958:
ERR_FS_XCHECK_PST
When both file systems describe the same file, the value of the attribute Protection System Type described by both file systems must be equal.

>>> [DVD] ODDITY 5959:
ERR_FS_XCHECK_FILE_NOT_FOUND
When either the [ISO] or the [ECMA]/[UDF] file system describes a file, the other file system must also describe that file.

9.4 DVD+RW VIDEO SPECIFIC CHECKS

9.4.1 Physical (DVD) Data Checks

9.4.1.1 Sector Header Checks

These checks relate to the 12 bytes sector ‘header’ data.

9.4.1.1.1 DVD+RW Video Specific Checks

[DVD+VR] ERROR **4991** (ref. [DVD+RW] 13.1.1)
ERR_DVDRW_SECTOR_ID
The sector Identification Data '<data field name>' is incorrectly <value>, while it must be <value>.

This is a message reporting an error in one of the sector header ID bits.

[DVD+VR] ERROR **4992** (ref. [DVD+RW] 2.2.1.1)
ERR_DVDRW_SECTOR_ID_B28
The sector Identification Data '<data field name>' is incorrectly <value>, while it must be <value> for all sectors containing VOB data.

This is a message reporting specifically an error in one of the sector header ID bit 28.

As of version 1.0 it is no longer required. So it is suppressed in v1.0 of the Verifier.

[DVD+VR] RECOMMENDATION VIOLATION **4993** (ref. [DVD+RW] 2.2.1.1)
ERR_DVDRW_SECTOR_ID_B28_DATA
The sector Identification Data '<data field name>' is set to <value>. It is recommended to be <value> for all DVD+RW Video defined data structures.

This is a message reporting specifically an error in one of the sector header ID bit 28.

As of version 1.0 it is no longer required. So it is suppressed in v1.0 of the Verifier.

[DVD+VR] ERROR **4995** (ref. [DVD+RW] 2.2.2)
ERR_DVDRW_SECTOR_RSV_CM
The sector RSV '<CGMS , APT>' has the value <value>;
It <shall be at most | must be> <value> <(other values reserved) | (reserved) when CPM is 0b>.

This is a spec v1.1 message reporting an error in the sector header RSV bits.

9.4.1.2 Lead-in Checks

These checks relate to the Lead-in data.

Additional to the ones defined by the DVD-ROM specification, DVD+RW also specifies some new data zones in the Initial Zone: Inner Disk Test Zone, Inner Drive Test Zone, Guard Zone 1 and Inner Disk Identification Zone.

9.4.1.2.1 DVD-ROM Generic Checks

[DVD+VR] ERROR **5010** (ref. [DVD-ROM] 1.5.13)
ERR_LEAD_RESERVED_NOT_ZERO
Lead-in Reserved bits have the value <value>; These should be all zero.

[DVD+VR] ERROR **5012** (ref. [DVD-ROM] 3.4.1.2, [DVD+RW] 17.9)
ERR_LEAD_REFCODE_ZONE_BAD
Lead-in Reference Code Zone sector <PSN> is not completely set to 0xAC.

[DVD+VR] ERROR **5014** (ref. [DVD-ROM] 3.4.1.5/6, [DVD+RW] 17.10/12)
ERR_LEAD_BUFZONE_NOT_NULL
Lead-in Buffer Zone sector <PSN> is not completely set to (00).

This is checked per sector and reports this error as soon as 1 byte is not zero, without specifying which byte(s) are actually non-zero.

[DVD+VR] ERROR **5016** (ref. [DVD-ROM] 3.4.1.3, [DVD+RW] 17.11)
ERR_LEAD_CTRL_DATA_DIFF
Lead-in Control Data Zone block <number> sector <PSN> is different from the corresponding sector of previous blocks.

This is checked per sector and reports this error as soon as 1 byte is different from the corresponding byte of the corresponding sector of previous Control Blocks, without specifying which byte(s) are actually different.

[DVD+VR] ERROR **5017** (ref. [DVD-ROM] 3.1.4)
ERR_LEAD_PFI_1ST_LAYER_END_ON16
Lead-in Control Data Zone Physical format information end address of the Dual layer, Opposite track path disc first layer is <value>;
The inverted address <value>, is no multiple of 16, as required.

[DVD+VR] ERROR **5018** (ref. [DVD-ROM] 3.4.1.3.1, [DVD+RW] 17.11.1)
ERR_LEAD_PFI_FIXED_VAL
Lead-in Control Data Zone Physical Format information field '<name>' has the value <value>; It can only have the fixed value <value>.

This error is generated for each PFI data field that has a value different from the fixed value specified in the standard.

[DVD+VR] ERROR **5019** (ref. [DVD-ROM] 3.4.1.3.1, [DVD+RW] 17.11.1)
ERR_LEAD_PFI_VAL_ILL
Lead-in Control Data Zone Physical format information field '<name>' has the illegal value <value>; It must be <at most | larger than><value>.

This error is generated for each PFI data field that has a value smaller or larger than allowed by the standard.

[DVD+VR] ERROR **5020** (ref. [DVD-ROM] 3.4.1.3.3, [DVD+RW] 17.11.3)
ERR_LEAD_CONT_PROV_NOT0
Lead-in Content provider info sector <relative number> is not completely set to (00).

For each of the 192 control data blocks, the last 14 sectors containing the Content provider information must completely be filled with zeroes

[DVD+VR] ERROR **5021** (ref. [DVD-ROM] 3.4.1.4 e.f., [DVD+RW] 17.1..4)
ERR_LEAD_ZONE_NOT_NULL
Lead-in < Inner Disk Test Zone | Inner Drive Test Zone | Guard Zone> sector <relative number> is not completely set to (00).

This is generated for any specific zone (number of sectors) of the Lead-in that is specified to contain only zeroes.

9.4.1.2.2 DVD Inherited Checks

[DVD+VR] ERROR **5022** (ref. [DVD-ROM] 3.4.1.3.1)
ERR_LEAD_PFI_TP_ILL
Lead-in Control Data Zone Physical format information 'Track path' field has the value <value> while the Number of layers is <value>;
It <"can only" | "must"> be <value> for a DL disc (Number of layers <value>).

- This is not relevant for DVD+RW Video, since it is basically a Single Layer only format.

[DVD+VR] ERROR **5023** (ref. [DVD-ROM] 3.4.1.3.1)
ERR_LEAD_PFI_LD_ILL
Lead-in Control Data Zone Physical format information 'Linear density' field has the value <value> while the Number of layers is <value>;
It <"can only" | "must"> be <value> for a %s Layer disc (Number of layers <value>).

- This is not relevant for DVD+RW Video, since it is basically a Single Layer only format.

[DVD+VR] ERROR **5025** (ref. [DVD-ROM] 3.4.1.3.1, [DVD+RW] 17.11.1)
ERR_LEAD_PFI_DATA_ZONE_STRT_ERR
Lead-in Control Data Zone Physical format information 'Start sector nr of the Data area' field has the illegal value <hex value> (<value>);
It must be set to 0x30000.

9.4.1.2.3 DVD+RW Video Specific Checks

DVD+RW defines 26 extra bytes of PFI data (cf. **[DVD+RW]** Table 9) for which the following additional checks have been added.

[DVD+VR] WARNING **5029** (ref. [DVD+RW] 14.4.2, 17.11.1, Annex L)
ERR_DVDRW_LEAD_PFI_FIXED_VAL_WARN
Lead-in Control Data Zone Physical format information field '<data field name>' has the value <hex value>; It should be <hex value>.

This message informs the user about a PFI data field having a value that is unexpected and probably unintended.

Currently this message can only result from a check on the PFI “number of PFI bytes in use” field, having the value 0. This actually indicates that the extra bytes are not used (yet), which is normal in some cases e.g. blank disc or a disc recorded with non-standard equipment, and as such is not illegal. Hence it is reported as a warning message.

[DVD+VR] ERROR **5030** (ref. [DVD+RW] 14.4.2, 17.11.1, Annex L)
ERR_DVDRW_LEAD_PFI_FIXED_VAL
Lead-in Control Data Zone Physical format information field '<data field name>' has the illegal value <hex value>; It can only be <hex value>. This message reports a PFI data field having an illegal value.

Currently this message can only result from a check on the PFI “number of PFI bytes in use” field, having another value than the default value 0x39 (57) or special case value 0.

[DVD+VR] ERROR **5031** (ref. [DVD+RW] 14.4.2, 26.3.2, G.2)
ERR_LEAD_EPS_OUT_RANGE
Lead-in Control Data Zone Physical format information field <name>' has the value <value>; It must be in the range [<low value>..<high value>].

This message reports one of the additional PFI data fields having an illegal value. For allowed values refer to the applicable sections in **[DVD+RW]**.

[DVD+VR] ERROR **5032** (ref. [DVD+RW] 14.4.2)
ERR_LEAD_PULS_OUT_RANGE
Lead-in Control Data Zone Physical format information field '<name>' has the value <value>; It must be in the range [<low value>..<high value>].

This message reports one of the additional PFI data fields having an illegal value. For allowed values refer to the applicable sections in **[DVD+RW]**.

[DVD+VR] ERROR **5035** (ref. [DVD+RW] 17.11.1, 19.1)
ERR_LEAD_PFI_DATA_END_ERR
Lead-in Control Data Zone Physical format information last recorded Physical Sector nr is <hex value>. It must be is <hex value> which is the first sector after the Data Zone of a fully formatted disc.

This check is only performed if the type of Lead-out is known (from the FDCB data) and it is described as being a complete Lead-out.

[DVD+VR] ERROR **5036** (ref. [DVD+RW] 21.3)
ERR_DVDRW_ENDDATA_GAP
The Temporary Lead-out must immediately follow the user data.

However the last recorded file <name> ends at sector is <value> (is <hex value>), while the PFI End Data Zone address points to is <hex value>, which creates a gap of is <value> sectors that is larger than 1 ECC block.

This is actually a cross check with the file system(s) data and is performed after the file systems have been parsed. It only uses the ISO-9660 file system data (Other cross checks will report if UDF data is not identical to ISO-9660 recorded data).

This check is only performed if the type of Lead-out is known (from the FDCB data) and it is described as being a Temporary Lead-out.

[DVD+VR] ERROR **5037** (ref. [DVD+RW] 17.11.1)
ERR_DVDRW_ENDDATA_ILL_COPY
Lead-in Control Data Zone block <number> PFI last recorded Physical Sector Number is <hex value>. Since it is no copy of the actual value recorded in the first 16 Control Data blocks, it must be <hex value>.

As off spec v1.1, it is no longer required (as is for DVD-Video) that all 192 instances of the Control Zone (ECC) blocks are identical. Now this is only mandatory for the first 16 instances. The remaining instances may be different in only one data field, i.e. the end address of the recorded data zone. But if it has not the same value as that in the first control block, it must specify the very last possible address, i.e. the address just before the Buffer Zone 3, 0x26053F.

9.4.2 Generic System Checks

[DVD+VR] ERROR **6001** (ref. [DVD+VR] 1.6)
ERR_DVDVR_SRSV_0
Reserved bits are <hex value>. These must be all 0.

[DVD+VR] SYSTEM ERROR **6009** (ref. N/A)
ERR_DVDVR_SYS_ILL_OPT_PARA
Command line parameter `<character>' not allowed for `-<character>' option!

The user has specified an illegal option parameter on the command-line.

[DVD+VR] SYNTAX ERROR **6010** (ref. N/A)
ERR_DVDVR_SYNTAX_RECOVER
<text string> : Parsing impossible due to syntax error : data skipped.

Only used by intermediate versions of the verifier program.

[DVD+VR] WARNING **6011** (ref. N/A)
ERR_DVDVR_NO_VTSI
No matching VTSI file found in the current directory !
Unless the VTSTT_VOBS contains only one VTS, parsing may be incorrect due to mixing presentation data from different VTSs , resulting in unjustified error messages.
In any case Buffer Cells and rubbish sectors between successive VOBs will be parsed, generating unjustified error messages.

This is a message issued at the start of a verification run if no VTSI file, not even the first and mandatory “VTS_01_1.IFO”, can be found in the current input directory. It warns the user about problems to be expected lacking the VTSI data and its Cell address table to control the parsing process.

[DVD+VR] WARNING **6012** (ref. N/A)
ERR_DVDVR_VTSI_MISSING
The VTSI file <name> for VTS <number> is missing in the current directory !
Unless the VTSTT_VOBS contains only one VTS, parsing may be incorrect due to mixing presentation data from different VTSs, resulting in unjustified error messages.
In any case Buffer Cells and rubbish sectors between successive VOBs will be parsed, generating unjustified error messages.

This is a message issued at the start of a verification run if the required VTSI file of the VTS with the specified number “VTS_<nr>_1.IFO” can’t be found in the current input directory. It warns the user about problems to be expected lacking the VTSI data and its Cell address table to control the parsing process.

[DVD+VR] WARNING **6015** (ref. N/A)
ERR_DVDVR_NO_VTSI_DATA
No Xcheck data for VTSI <number> file present in the Xcheck data file !
Either the VTSI has not been parsed yet, or no VTSI file is available.
Unless the VTSTT_VOBS contains only one VTS, parsing may be incorrect due to mixing presentation data from different VTSs, resulting in unjustified error messages.
In any case Buffer Cells and rubbish sectors between successive VOBs will be parsed, generating unjustified error messages.

This is a message issued before verification of a file is actually started if the required VTSI (Cell address table) data can not be found in the cross check data file. If the cross check data file is not corrupted, it might be caused by the fact that the VTSI file has not been parsed yet. It warns the user about problems to be expected lacking the VTSI data and its Cell address table to control the parsing process.

[DVD+VR] INFORMATION **6021** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_UNK_FNAME
The specified input file name, '<file name>', is not compliant with the DVD+VR specification! Therefore the derived domain, '<domain name>', may be not correct.

If the specified input file name does not start with "VTS_<xx>_<y>" (with x,y = 0..9) or "VIDEO_TS", the verifier assumes the DVD Domain to be the Title Domain. This message is reported to warn the user that this assumption may be incorrect resulting in some unjustified error messages.

[DVD+VR] INFORMATION **6025** (ref. N/A)
ERR_DVDVR_JUMP_START
Jumping to <sector/pack RLBN number> in file <file name>

This is a purely informative message to the user notifying him about the initial jump to the (non-zero) verification start position conform the user input. It is the result of having specified a non-zero verification start position.

[DVD+VR] INFORMATION **6026** (ref. N/A)
ERR_DVDVR_JUMP_BEYOND
Jump beyond file size <value>

This is a purely informative message to the user explaining why parsing of a complete file is skipped: because the initial verification start position is not within the current VTS file but in one of its successors. It is the result of having specified a non-zero verification start position.

[DVD+VR] ERROR **6028** (ref. N/A)
ERR_DVDVR_ERR_FSIZE
Unable to derive the file (<name>) size (fsys_fsize) !

This message is generated when the verifier "fsys_fsize" routine is unable to return a correct file size (in sectors) value derived from the earlier parsed file systems data for the specified file.

[DVD+VR] INFORMATION **6029** (ref. N/A)
ERR_DVDVR_SKIP_BEYOND
Next <Cell | chapter> beyond the current VTS file size <value>.

This is a purely informative message to the user reporting that skipping sectors of Cells not belonging to the current VTS or not part of the VR Play List, will take the parser into one of the next (successor) files of the current VTS. It is a result of the parser control mechanism (cf.) selected by the user.

[DVD+VR] SYSTEM ERROR **6031** (ref. N/A)
ERR_DVDVR_BAD_IFO_USE_BUP
The VRMI data IFO file is ignored for cross checking and the BUP data used instead, because <the IFO file is probably corrupt | the user forced use of the BUP file>.

This message informs the user that the original VRMI data (on the “VIDEO_RM.IFO” file) will not be used for cross checking with other VMGI, VTSI or VOBS data, but the backup data will be used instead. This can be caused for any of the specified reasons:

1. The IFO file has been judged to be corrupt (because it is marked as such, i.c. its size is 1 byte, or it has a too low (illegal) size)
2. The user has specified to (forcedly) use the backup data

Note that this message is the DVD+RW Video version of DVD ERROR 3003.

[DVD+VR] SYSTEM ERROR **6032** (ref. N/A)
ERR_DVDVR_BAD_IFO_AND_BUP_USE_NONE
The VRMI data IFO and BUP files are ignored for cross checking because neither is reliable.

This message informs the user that both the original and backup VRMI data (on files “VIDEO_RM.IFO” and “VIDEO_RM.BUP”) will not be used for cross checking with other VMGI, VTSI or VOBS data. So effectively cross checking will be disabled! This can be caused because both files, having a too low illegal size, are judged to be corrupt.

Note that this message is the DVD+RW Video version of DVD ERROR 3004.

[DVD+VR] SYSTEM ERROR **6035** (ref. N/A)
ERR_DVDRW_POSSIBLE_BAD_SPOT
Possible 'Bad Spot' detected on the disc!
Disc may be corrupt. <The parsing process is reset | Verification is aborted>.

This message can only be generated while verifying an actual DVD+RW Video disc and informs the user that possibly a “bad spot” has been hit on the disc (cf. 12 Defective Media Handling). This may prevent further reliable parsing and verification of the disc’s data and the specified action has been taken to avoid serious verification problems.

[DVD+VR] INFORMATION **6040** (ref. N/A)
ERR_DVDVR_SKIP_CELL
At pack <number> : Skipping <Cell not belonging to this VTS | Chapter not part of this VR Play List >...

This is a purely informative message notifying the user that the parser is about to start (from the specified RLBN on) skipping sectors not belonging to Cells of the current VTS or not part of the VR Play List. It is a result of the parser control mechanism (cf.) selected by the user.

[DVD+VR] INFORMATION **6041** (ref. N/A)
ERR_DVDVR_PROC_CELL
At pack <number> : Resumed parsing <Cell after interleaved Cell data | after VR Play List jump> ...

This is a purely informative message notifying the user that the parser is again resuming (from the specified RLBN on) parsing sectors belonging to Cells of the current VTS after having skipped some sectors, or sectors part of the VR Play List, after having jumped over a hidden chapter. It is a result of the parser control mechanism (cf.) selected by the user.

[DVD+VR] INFORMATION **6045** (ref. N/A)

ERR_DVDVR_BUFF_CELL

At pack <number> : Buffer Cell with C_IDN=<ID number> detected...

This is a purely informative message notifying the user that the parser has detected a Buffer Cell with the specified ID number. It will not be parsed.

9.4.3 VOBS Data Checks

9.4.3.1 DVD Application Checks

9.4.3.1.1 VOB Checks

[DVD+VR] INFORMATION **6050** (ref. [DVD+VR] 3.2.3)
ERR_DVDVR_VOB_START_MISSING
Some packs are missing from the start of the current VOB!

This is actually not an error but simply reported as an information message.

Missing packs at the VOB start are detected by comparing the start time of the VOB and its first VOBU: If these are not equal then at least the NAV_PCK of the VOBU preceding the current first VOBU is missing.

[DVD+VR] ERROR **6053** (ref. [DVD+VR] 3.2.3.2)
ERR_DVDVR_VOB_LAST_CELL_NO_BUF
The last Cell of the current VOB is not a Buffer Cell since there is not even a valid NV_PCK.

The first pack of the Cell following the last valid, playable Cell of a VOB must be the NV_PCK of a Buffer Cell. It is checked for a few basic NV_PCK requirements, i.e. the presence of a system_header and private_stream_2 start code and PCI sub_stream_id on the expected locations. If these are not fulfilled, this error message is generated.

[DVD+VR] ERROR **6054** (ref. [DVD+VR] 1.5.2)
ERR_DVDVR_VOB_BUF_CELL_ERR
The last Cell of the current VOB is not a valid Buffer Cell:
Its C_IDN is <value>, which is not different from the preceding Cell.

The first pack of the Cell following the last valid, playable Cell of a VOB must be the NV_PCK of a Buffer Cell. Its C_IDN number form its DSI is compared with the C_IDN of the preceding (last playable) Cell of the VOB. If both are not different, this error message is generated.

[DVD+VR] ERROR **6059** (ref. [DVD+VR] 3.2.3.2)
ERR_DVDVR_VOB_NO_TOP_FLD_START
The first video frame in display order of the VOB (possibly missing some packs at the start) does not start with a top field.

This is checked at the first picture of the first VOBU of a VOB, by checking whether either it has TOP_FIELD picture_structure or it is a FRAME_PICTURE and has its top_field_first set.

9.4.3.1.2 Cell Checks

[DVD+VR] ERROR **6061** (ref. [DVD+VR] 3.2.3.2 Annex-D-1)
ERR_DVDVR_ CELL_PBTM_EXCEEDED
The Cell Playback time C_PBTM <time value> = <float value> seconds, specified by Cell <number> of PGCI <number> exceeds the max. allowed value <float value> for <CBR or CVBR | non-specified> bit rate.

This message reports a Cell's Playback Time exceeds the max allowed value as specified in Annex D.1 of the [DVD+VR] spec.

Since it uses the bit rate (BR) specified in the VRMI, it can be considered a cross check.

9.4.3.1.3 VOBU Checks

[DVD+VR] ERROR **6066** (ref. [DVD+VR] 3.2.3.2)
ERR_DVDVR_VOBU_NO_VIDEO_FRAME
The current VOBU does not contain at least one coded video frame.

This is simply checked by inspecting at the end of a VOBU if at least one picture end (EVT_PICTURE_END) has been encountered.

9.4.3.1.4 VOBS Boundary Detection Messages

These are all, mainly informative, warning messages generated by the verifier's VOBS boundary detection module. They indicate a problem encountered in the detection process, which may not be caused by a non-conformance with any of the DVD+RW Video specifications.

[DVD+VR] WARNING **6071** (ref. [DVD+VR] N/A)
ERR_DVDVR_VOBU_NO_END_DETECT
No valid VOBU end detectable!

This message is issued by the VOBU end detection module to warn that it is incapable of properly detecting the end of the VOBU.

Unlike DVD-Video, a VOBU's end no longer has to coincide with the start of the next VOBU. There could be 'gaps' in between with garbage or data of other VTSs. So a real VOBU end detector is active using the VTSI recorded VOBU start address table. The end of a VOBU is found when the current pack number is one less than the start of the next valid VOBU. If the VOBU address table is not available (e.g. missing cross check data), the current or next VOBU can not be found in the table, this message is generated.

This message is 'normal' at the end of a VTS, when accessing the last VOBU, since then there simply is no next VOBU recorded in the address table.

[DVD+VR] WARNING **6072** (ref. [DVD+VR] N/A)
ERR_DVDVR_VOBU_END_FORCED
VOBU end event forced!

If the start of a VOBU is detected but the end of the previous VOBU has not been flagged by a VOBU end event, this event is forced before issuing a start event for the next VOBU and it is reported to the user by this message. This is done to prevent all kinds of problems because of that 'unclosed' VOBU.

[DVD+VR] WARNING **6073** (ref. [DVD+VR] N/A)
ERR_DVDVR_CELL_NO_END_DETECT
No valid Cell end detectable!

The end of a Cell is detected using the VTSI recorded Cell address table. A Cell end is signaled when the current pack number matches the (RLBN) end address of the Cell recorded in the VTS_C_ADT. If the Cell address table is not available (e.g. missing cross check data) or the current Cell, i.e. the Cell the current pack belongs to, can not be found in the table, this message is generated.

[DVD+VR] WARNING **6074** (ref. [DVD+VR] N/A)
ERR_DVDVR_CELL_END_FORCED
Cell end event forced!

If the start of a Cell is detected but the end of the previous Cell has not been flagged by a Cell end event, this event is forced before issuing a start event for the next Cell and it is reported to the user by this message. This is done to prevent all kinds of problems because of that 'unclosed' Cell.

[DVD+VR] WARNING **6075** (ref. [DVD+VR] N/A)
ERR_DVDVR_VOB_NO_END_DETECT
No valid VOB end detectable!

The end of a VOB is detected using the VTSI recorded Cell address table. A VOB end is signaled when the current pack number matches the (RLBN) end address of the current Cell as it is recorded in the VTS_C_ADT, and if the next Cell does not start at the next pack. In other words, if there is a 'gap' between the current Cell and the next. If the Cell address table is not available (e.g. missing cross check data) or the current Cell, i.e. the Cell the current pack belongs to, can not be found in the table, this message is generated.

[DVD+VR] WARNING **6076** (ref. [DVD+VR] N/A)
ERR_DVDVR_VOB_END_FORCED
VOB end event forced!

If the start of a VOB is detected but the end of the previous VOB has not been flagged by a VOB end event, this event is forced before issuing a start event for the next VOB and it is reported to the user by this message. This is done to prevent all kinds of problems because of that 'unclosed' VOB.

9.4.3.2 MPEG System Checks

The module should perform all checks on the MPEG PS system data of a DVD+RW Video disc's presentation layer.
These checks verify conformance with additional DVD+RW Video specific constraints on MPEG data structures such as pack, system_header or PES_packets.

The required checks, as derived from the DVD+RW Video spec **[DVD+VR]** are listed below. Unless explicitly stated otherwise, these are all reported as errors.

9.4.3.2.1 Generic PS Checks

None.

9.4.3.2.2 Pack Checks

[DVD+VR] ERROR **6101** (ref. [DVD+VR] 3.2.3)
ERR_DVDVR_SCR_OUT_RANGE
The current pack's SCR value <value> is not within the valid range 0.. <value> (VOB_V_E_PTM).

The value actually used is the VOB end time value which is stored only at each EVT_DSI event. As a consequence, this value still has its initial zero value when the first pack's SCR is checked. But since :

- the VOB_V_E_PTM value does not change within one VOB
- SCR values must increase continuously within a VOB

Then if the first SCR is wrong (while not reported) the next one definitely is illegal too, and this will be reported.

[DVD+VR] ERROR **6103** (ref. [DVD+VR] 3.2.3)
ERR_DVDVR_SCR_0_NOT_1ST
The SCR of the VOB's first pack is zero while it is not the first pack of both a VOB and a Cell.

It is not allowed to have the SCR reset to zero if not at the start of a VOB or Cell.

To detect the start of the current VOB the start address value stored upon a EVT_VOB_START event is compared with the current pack number.

[DVD+VR] ERROR **6105** (ref. [DVD+VR] 3.2.3)
ERR_DVDVR_SCR_NOT0_TOO_SMALL
The current pack's SCR value <value> should be at least <value> when some packs are missing from the start of the current VOB.

The minimal allowed value for any SCR is 36000 (in 90000 kHz tick units) (which is the minimal presentation time, 0.4 sec, of a VOBU).
In most cases the first SCR will be non-zero only when there are some packs missing, because usually a VOB starts with a zero SCR. However in DVD+RW Video this is only recommended and not mandatory!
To detect the start of the current VOB the start address value "start_add " stored upon a EVT_VOB_START event is compared with the current pack number.
Whether some packs are indeed missing from the start of the current VOB, is not taken into account for this check: It is considered not relevant.

Disabled DVD Checks

In the DVD specific MPEG pack verification module changes have been made to disable or replace the checks listed below. In case of replacement by a DVD+RW Video variant or to be active for the VMGM_VOBS only, these are implemented or simply copied to a DVD+RW specific module.

[DVD] ERROR **3107** (ref. [DVD-3] 3.3.12.4)
ERR_DVD_SCR_0
SCR in first pack is <value>; SCR in the first pack of each VOB must be 0.

9.4.3.2.3 System_header Checks

None.

9.4.3.2.4 PES Checks

[DVD+VR] ERROR **6151** (ref. [DVD+VR] 3.2.3)
ERR_DVDVR_PSTD_AFTER_1ST
P-STD parameters are encoded in a PES_packet header after the first VOBU!

Since the only allowed PES_extension data in DVD or DVD+RW Video are the P-STD parameters, this is checked by signalling whenever the PES_extension_flag is set and the packet does not belong to the first VOBU of the VOB.
A 2nd check is done by inspecting the PES_extension P_STD_buffer_flag value, which may not be set (although it should not be present at all).

Note that this may look as a relaxation of the DVD-Video constraint that only the FIRST PES_Packet may have P-STD parameters encoded. But it is not, because Sub-picture packets and probably also Audio packets can start well after the first VOBU!

Because the P-STD parameters can be absent, the STD buffer model will remain disabled. Since the buffer sizes are fixed in DVD+VR, a function will be called by the EVT_PES_PACKET_HEADER (mpeg_data) event and will set the buffer sizes for the P-STD buffer model.

Disabled DVD Checks

In the DVD specific MPEG PES_packet verification module changes have been made to disable or replace the checks listed below. In case of replacement by a DVD+RW Video variant or to be active for the VMGM_VOBS only, these are implemented or simply copied to a DVD+RW specific module.

[DVD] ERROR **3207** (ref. [DVD-3] Table 5.2.3-1 e.f. Note 2)
ERR_DVD_PKT_PES_MISSING
PES_packet <number> has no (P_STD_buffer_size) PES_extension
(expected for the first <Video | Audio | Sub-picture> packet of a VOB).

For DVD+RW Video, this only holds for the first VOBU of a VOB.

Disabled MPEG Checks

[DVD] ERROR **1430** (ref. MPEG Systems 2.7.7)
ERR_NO_BUFSIZE
No STD_buffer_size in first packet of <video | audio> stream <stream number>.

9.4.3.3 SPU Checks

[DVD+VR] ERROR **6181** (ref. [DVD+VR] 3.3.8)
ERR_DVDVR_SPU_NOT_IN_1_VOBU
All data of the current SPU is not contained in 1 VOBU.

This is verified at the end of a VOBU, by inspecting two flags defined in the DVD+RW Video VOBU verification object. Both, flag “has_spu”, indicating that SPU data is present in the VOBU and flag “spu_completed”, signalling that the end of the current SPU data has been parsed, must have been set.

[DVD+VR] ERROR **6182** (ref. [DVD+VR] 3.3.8)
ERR_DVDVR_SPU_NOT_CONTAINED
The current SPU with presentation start/end time <value>, is encoded in the current VOBU with presentation start/end time <value>, where it does not belong to.

This check signals when a SPU is encoded in a VOBU during which presentation it is not displayed at all (not even overlaps), i.e. when the SPU_PST is larger than the VOBU end time, or the SPU_PTT smaller than the VOBU start time.

[DVD+VR] ERROR **6183** (ref. [DVD+VR] 3.3.8)
ERR_DVDVR_SPU_VALID_PERIOD
The current SPU's validity period <start | end> time <value>, is <smaller | larger> than the VOBU <start | end> PTM <value>.

This check flags when a SPUs presentation time surpasses the VOBU display time, either because it starts earlier or terminates later.

9.4.3.4 Elementary Stream Checks

This chapter lists all checks on the MPEG ES data and more specifically the Video & Audio data present on a DVD+RW Video disk.

These checks fall roughly into 3 distinct groups:
1. Checks on DVD+RW Video specific constraints on MPEG data
2. Checks on DVD+RW Video specific constraints on DVD data
3. Checks on DVD+RW Video specific data

The required checks, as derived from the DVD+RW Video spec **[DVD+VR]**, are listed below. Unless explicitly stated otherwise, these are all reported as errors.

9.4.3.4.1 Video Checks

[DVD+VR] INFORMATION **6250** (ref. [DVD+VR] 3.2.1 Table 3-1)
ERR_DVDVR_SET_MPEG1
Sequence_header specifies a low resolution vertical_size value <value>. Video coding mode is set to MPEG-1.

DVD+RW Video demands MPEG-1 video encoding for the 'low' resolution LP recording mode. When it is not specified that the current VTS uses MPEG-1 LP mode, it is explicitly set by the verifier when detecting the low vertical resolution setting in the video sequence_header and this message is generated to inform the user. However a 'low resolution, LP' VTS can be specified either by one of the MPEG-1 input stream verifier command-line or script file options, or by the corresponding VTSI Cross Check data.

This is reported as an information message.

[DVD+VR] ERROR **6251** (ref. [DVD+VR] 3.2.1 Table 3-1)
ERR_DVDVR_HVSIZE_ILL
Sequence_header : illegal <horizontal | vertical>_size value <value>.

The sequence_header specifies a video horizontal or vertical size value that is not supported by DVD+RW Video. Actually currently only the illegal horizontal size value 704 has to be detected, which is the only (additional) value that is not allowed for DVD-Video.

[DVD+VR] ERROR **6253** (ref. [DVD+VR] 3.2.1 Table 3-1)
ERR_DVDVR_VSIZE_NMPG1
Sequence_header specifies an MPEG-2 vertical resolution <value>, while the video data is MPEG-1.

When the video encoding mode of the current VTS is known to be MPEG-1 (because it is specified as such by the user or by VTSI Cross Check data), only the low SIF vertical resolution can be used, i.e. 288 or 240 for resp. PAL or NTSC.

9.4.3.4.2 Audio Checks

There are no checks on Audio ES (MPEG or AC3), only relaxations to the DVD-Video rules (on navigation data), defined in [DVD+VR] 3.2.7.
Section 3.4.3 of the DVD+RW Video spec deals with IEC-60958 audio, which is not supported (yet).

9.4.4 Physical (DVD+RW) Data Checks

9.4.4.1 Generic

[DVD+VR] ERROR **6300** (ref. N/A)
ERR_LEAD_RSRVD_BITS
Lead-<in | out> <data field name> reserved bits contain the value <hex value>. These must be all zero.

[DVD+VR] INFORMATION **6305** (ref. [DVD+RW] 21.3)
ERR_LEAD_DISC_FORMAT_INFO
According to the recorded FDCB info, the current disc(image) < “is Partly Formatted” | “is Fully Formatted” | “formatting process is still active and FDCB possibly not up to date !”>.

This is a purely informative message reporting the formatting status of the disc under test..

[DVD+VR] WARNING **6306** (ref. [DVD+RW] 21.3)
ERR_LEAD_DISC_FORMAT_WARN
According to the recorded FDCB info, the current disc(image) <”is not formatted ! “ | “is not properly formatted (missing an FDCB) !”>; This may be unintended.

This message warns the user about a possible incorrectly formatted disc.

[DVD+VR] ERROR **6307** (ref. [DVD+RW] 21.3, [DVD+VR] 2.1)
ERR_LEAD_DISC_FORMAT_ERR
According to the recorded FDCB info, the current disc(image) <”has Intermediate Format !” | “is not completely formatted (interrupted ?).”>. This is not a legal DVD+RW Video format.

This message reports formatting problems with the disc under test. DVD+RW Video only supports Fully or Partly Formatted discs.

9.4.4.2 DMA Zone and RPL Checks

As of specification version 1.0, Defect Management is no longer part of the DVD+RW specification and the DMA Zone incl. its RPL data has been removed from the Lead-in and Lead-out. As a consequence the verifier no longer supports the related checks and error messages described below.

DMA Zone

[DVD+VR] ERROR **6311** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_INIT_NO_DRL
Lead-<in | out> DMA Zone <number> has no DRL block recorded, but initially the first block of each DMA Zone shall be recorded with a DRL.

[DVD+VR] ERROR **6312** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_INIT_NOT0
Lead-<in | out> DMA Zone <number> DRL block <number> has not all its sector data set to 0, but initially all but the first block of each DMA Zone must have all Main data bytes set to 0.

[DVD+VR] ERROR **6314** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_DRL_NOTEQ
Lead-<in | out> DMA Zone <number> DRL <number> <field name> has the value <value> which is different from <value> of DRL <number>. But the valid DRL of each DMA Zone must contain the same data.

[DVD+VR] ERROR **6317** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_RSRVD_SECTOR
Lead-<in | out> DMA Zone <number> DRL <number> is a DRL spare block; However not all Main Data bytes are zero.

RPL Data

[DVD+VR] ERROR **6321** (ref. [DVD+RW] XXX)
ERR_LEAD_RPL_ID_ERR
Lead-<in | out> DMA Zone <number> DRL specifies an illegal DRL Identifier <characters>; It must be 'DRL'.

[DVD+VR] ERROR **6322** (ref. [DVD+RW] XXX)
ERR_LEAD_RPL_VERS_ERR
Lead-<in | out> DMA Zone <number> DRL specifies an illegal Version_number <value>; It must be set to 01h.

[DVD+VR] ERROR **6324** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_FMT_COUNT
Lead-<in | out> DMA Zone <number> DRL Update Count specifies the number <value>. This must be set to 0 during the formatting operation.

[DVD+VR] ERROR **6326** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_N_RPL_ERR
Lead-<in | out> DMA Zone <number> DRL specifies an illegal N_RPL value <value>; It must be zero or in the range [510..4094].

[DVD+VR] ERROR **6327** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_NO_RPLS
Lead-<in | out> DMA Zone <number> DRL specifies the N_RPL value <value>; However linear replacement shall not be applied and N_RPL must be zero.

- This is a DVD+RW Video specific constraint on the DVD+RW generic DMA data structure.

[DVD+VR] ERROR **6331** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_STAT2_NOT0
Lead-<in | out> DMA Zone <number> DRL entry <number> is a DFT entry. However its Status 2 field is set to <hex value>, but must be set to 0.

[DVD+VR] ERROR **6332** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_DFT_BLOCKID_NOT0
Lead-<in | out> DMA Zone <number> DRL entry <number> is a DFT entry. However its Replacement Block ID is set to <hex value>, but must be set to zero.

[DVD+VR] ERROR **6333** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_DUP_BLOCKID
Lead-<in | out> DMA Zone <number> DRL <RPL | DFT> entry <number> has a Replacement Block ID <value> that is the same as its Defective Block ID.

[DVD+VR] ERROR **6335** (ref. [DVD+RW] XXX)
ERR_LEAD_DMA_DRL_NOT_SORTED
Lead-<in | out> DMA Zone <number> DRL entries are not sorted in ascending order: <RPL | DFT> entry <number> has <data field name> with value <value> preceding <RPL | DFT> entry <number> with <data field name> value <value>.

9.4.4.3 Disk Identification Zone and FDCB Checks

These checks relate to the Disk Identification Zone data in both the Lead-in and Lead-out.

Generic

[DVD+VR] ERROR **6361** (ref. [DVD+RW] 22.1..2)
ERR_LEAD_DCB_RSRVD_SECTOR
Lead-<in | out> Disc Identification Zone DCB <number> <remaining | reserved> sector <number> bytes are not all set to 0x0.

Reports if a DCB reserved bytes 'block' or some of the DCB's reserved sectors are not completely set to zero, for a FDCB as well as unused DCBs.

[DVD+VR] ERROR **6363** (ref. [DVD+RW] 22.2)
ERR_LEAD_IDZONE_FDCB_MISSING
Lead-<in | out> Disc Identification Zone has no Formatting DCB.

DVD+RW demands an FDCB to be present on a disc.

[DVD+VR] ERROR **6364** (ref. [DVD+RW] 22.2)
ERR_LEAD_IDZONE_MULTI_FDCB
Lead-<in | out> Disc Identification Zone has an additional FDCB in DCB <number>.

DVD+RW allows for only one FDCB to be present on a disc.

[DVD+VR] ERROR **6365** (ref. [DVD+RW] 22.1)
ERR_LEAD_IDZONE_DCB_NOTAT1ST
Lead-<in | out> Disc Identification Zone has a DCB encoded in DCB <number>. It must be written at the first unused DCB which is <number>.

DCBs have to be used 'sequentially', i.e. the DCB with the lowest number must be used first. Following an unused DCB, no DCBs can be used. This implies in case of a DVD+RW disc that effectively only the first DCB may be used (as an FDCB). This message reports the use of another DCB than the first available one.

[DVD+VR] ERROR **6366** (ref. [DVD+RW] 22.1)
ERR_LEAD_IDZONE_DCB_NOT_0
Lead-<in | out> Disc Identification Zone has an unused DCB encoded in DCB <number>, but the subsequent DCB <number> is also in use (Content Descriptor <hex value>).

Following an unused DCB, no DCBs can be used. This message reports the use of another DCB after a DCB specified as being unused.

DCB Data

[DVD+VR] ERROR **6370** (ref. [DVD+RW] 22.1)
ERR_LEAD_DCB_RSRVD_CONT_DESC
Lead-<in | out> Disc Identification Zone DCB <number> specifies a reserved Content Descriptor value <hex value>.

Only the values 0x00000000 (unused), 0x46444300 (FDCB), 0xFFFFFFFE (bad) and 0x FFFFFFFF (re-usable) are allowed for the DCB Content Descriptor.

FDCB Specific

[DVD+VR] ERROR **6381** (ref. [DVD+RW] 22.2)
ERR_LEAD_FDCB_BAD_VAL
Lead-<in | out> Disc Identification Zone FDCB <Unknown Content Descriptor Actions bit field name> field specifies the value <value>. It must be <value>.

The FDCB Unknown Content Descriptor Actions bytes must have the fixed value 0x0000000D, meaning the DCB shall not be modified, the disc shall not be reformatted and recording shall not be allowed in the Data Zone.

[DVD+VR] ERROR **6382** (ref. [DVD+RW] 22.2)
ERR_LEAD_FDCB_BAD_ADD
Lead-<in | out> Disc Identification Zone FDCB <Last Written Address | Last Verified Address | Bitmap Start Address | Bitmap Length> is <hex value>. It must be <larger than | at least | > <hex value> < | if the Formatting Bitmap is not used.

Reports violations of:

- Last Written Address > 0x30000
- Last Verified Address > 0x30000
- Bitmap Start Address != 0 and Bitmap Length != 0 if the Formatting bitmap is not used.

[DVD+VR] ERROR **6385** (ref. [DVD+RW] 17.11.1, 22.2)
ERR_LEAD_FDCB_PFI_ERR
Lead-<in | out> Disc Identification Zone FDCB <Last Written Address | Last Verified Address> is <hex value>. It must be larger than <hex value>, the last sector of the Data Zone as specified by the Lead-in PFI data.

This is actually kind of a cross checks between the Lead-in PFI and DIZ-FDCB data. The Last Written Address as specified by the FDCB must be larger than the Data Zone end address specified by the Lead-in PFI, since it also takes the disc's Lead-out data into account.

Since there is no longer Defect Management specified from version 1.0 onwards, the Last Verified Address is never modified and remains 0x30000. Its value is no longer checked

[DVD+VR] ERROR **6388** (ref. [DVD+RW] 21.3)
ERR_LEAD_FDCB_BITMAP_USED
Lead-<in | out> Disc Identification Zone Formatting DCB Formatting bitmap is being used (DCB sector <number> is not all zero); But it shall not be used for partially formatted discs.

[DVD+VR] ODDITY **6389** (ref. [DVD+VR] (2.2.2))
ERR_LEAD_FDCB_BITMAP_NOT_0
Lead-<in | out> Disc Identification Zone Formatting DCB Formatting bitmap (DCB sector <number>) is not all zero; But it should not be used.

- This was a DVD+RW Video v0.9 specific constraint on the DVD+RW generic FDCB data structure, and is no longer required as of v1.0. It is replaced by a comparable but more relaxed requirement in the DVD+RW basic format spec and as such covered by ERROR 6388. As a result of this, the error has changed into an oddity.

9.4.4.4 Lead-out Checks

These checks relate to the Lead-out data.

DVD+RW specifies some data zones in the Lead-out Zone comparable as those in the Lead-in: Outer Disk Test Zone, Outer Drive Test Zone, Guard Zone 2 & 3 and Outer Disk Identification Zone.

[DVD+VR] INFORMATION **6400** (ref. N/A)
ERR_LEADOUT_SKIP_NO_FDCB
Lead-out parsing skipped since its presence or location is unknown, because no FDCB has been found.

This is purely an informative message, notifying the user about the incapability of the verifier to properly locate the Lead-out, lacking the FDCB information whether it is actually recorded on disc and if so, if it is a temporary Lead-out recorded in the Data Zone or a Partial or Full Lead-out recorded in the Lead-out Zone.

[DVD+VR] ERROR **6410** (ref. [DVD+RW] 19.1)
ERR_LEADOUT_BUFZONE_NOT_NULL
Lead-out Buffer Zone 3 sector <PSN address> is not completely set to 0x0.

[DVD+VR] ERROR **6421** (ref. [DVD+RW] 21.3)
ERR_TEMP_LEADOUT_NOT_NULL
Temporary Lead-out ECC block <value>, sector <value> (PSN <hex value>) is not completely filled with all (00).

The Temporary Lead-out must be completely filled with zeroes or optionally may contain a (copy of) temporary Outer Disc Identification Zone in ECC blocks 48..63.
This message is reported if :

- not all sectors of the Temporary Lead-out first 48 ECC blocks are set to zero
- or the 1st sector of ECC block 48 does not start with an FDCB Content descriptor (which is required if it marks the start of an Outer Disc Identification Zone) and one of the other ECC blocks is not completely filled with zeroes

[DVD+VR] ERROR **6423** (ref. [DVD+RW] 21.3)
ERR_TEMP_LEADOUT_NOT_IN_DATA_ZONE
The temporary Lead-out is not completely within the Data Zone, since the FDCB Last Written Address, indicating the end of the Lead-out, is <hex value> which is outside the Data Zone (ending at <hex value>).

Since the temporary Lead-out must be completely located within the Data Zone, its end address as recorded in the FDCB Last Written Address can not be larger than the Lead-out Buffer Zone 3 start address at 0x260540, being the start of a Full Lead-out.

[DVD+VR] ERROR **6425** (ref. [DVD+RW] 21.3)
ERR_TEMP_LEADOUT_TOO_SMALL
The temporary Lead-out shall be at least 64 ECC blocks large. However only <number> blocks have been recorded. (FDCB Last Written Address <hex value> - PFI last Data Zone Sector <hex value>)

The size of the temporary Lead-out computed by subtracting its start address as specified by the Lead-in PFI Data Zone end address from its end address as recorded in the FDCB Last Written Address, must be at least 64 ECC blocks or 1024 sectors.

9.4.4.5 Lead-in vs. Lead-out Cross Checks

The following messages may be generated as result of a cross check error between a DVD+RW disc's Lead-in and Lead-out data.

[DVD+VR] ERROR **6451** (ref. [DVD+RW] 22.1..2)
ERR_LEAD_ODZONE_DCBS_MISS
Lead-in Inner Disc Identification Zone contains a FDCB in DCB <number> which is not found in the Lead-out Outer Disc Identification Zone.

The FDCB in Lead-in and Lead-out Disc Identification Zones must be identical. This error is reported when a DCB of the same type can not be found in both the Lead-in and Lead-out Identification Zone.

[DVD+VR] ERROR **6452** (ref. [DVD+RW] 22.1..2)
ERR_LEAD_IDZONE_DCBS_DIF_VAL
Lead-in Inner Disc Identification Zone FDCB in DCB <number> is different from the Lead-out version in DCB <number>; field '<name>' is resp. <hex value> and <hex value>.

The FDCB in Lead-in and Lead-out Disc Identification Zones must be identical. This message reports a different value in the specified numerical FDCB data field.

[DVD+VR] ERROR **6453** (ref. [DVD+RW] 22.1..2)
ERR_LEAD_IDZONE_DCBS_DIF_STR
Lead-in Inner Disc Identification Zone FDCB in DCB <number> is different from the Lead-out version in DCB <number>; field '<name>' is resp. '<string value>' and '<string value>'.

The FDCB in Lead-in and Lead-out Disc Identification Zones must be identical. This message reports a different value in the specified FDCB string data field.

9.4.4.6 Other messages

The following messages are purely informative and report the 'bit settings' (cf. **6.1 dump Bit Settings**) at some critical locations of a DVD+RW disc. There are multiple variants used to report the settings in Lead-in or Data Zone, or with or without Lead-in settings.

[DVD+VR] INFORMATION **6491** (ref. N/A)
ERR_BIT_SETTINGS_L
Encoded bit settings at sector PSN $<hex value> (<value>) : $<hex value> / <value> / <value>.

[DVD+VR] INFORMATION **6492** (ref. N/A)
ERR_BIT_SETTINGS_D
Encoded bit settings at sector PSN $<hex value> (<value>; LSN <value>) : $<hex value / <value> / <value>.

[DVD+VR] INFORMATION **6493** (ref. N/A)
ERR_BIT_SETTINGS_XL
Encoded bit settings at sector PSN $<hex value> (<value>) : $<hex value> / xx / xx.

[DVD+VR] INFORMATION **6494** (ref. N/A)
ERR_BIT_SETTINGS_XD
Encoded bit settings at sector PSN $<hex value> (<value>; LSN <value>) : $<hex value> / xx / xx.

9.4.4.7 ADIP Checks

These checks relate to the ADIP information coded in the Lead-in Zone wobble.

Checking this data is currently not supported and it is not clear yet whether verification of this data will actually be possible in the future, since it relies on having access to the wobble data for which a special drive (with e.g. a +RW basic engine) is needed. This is not supported by the current ASALE-made verification drive.

9.4.5 Navigation Data Checks

9.4.5.1 DVD+RW Video Specific VMGI Checks

[DVD+VR] ERROR **6501** (ref. [DVD+VR] 3.3.1)
ERR_DVDVR_VMGI_PARENTAL_MANAGEMENT
VMGI: No Parental management is allowed, but the PLT_MAIT_SA specified <hexadecimal value>

No parental management is applicable for DVD+RW Video, so the PTL_MAI Table is not allowed in the VMGI.

[DVD+VR] ERROR **6502** (ref. [DVD+VR] 3.3.2)
ERR_DVDVR_VMGM_PGCI_UT_SIZE
The size of the VMGM_PGCI_UT is <value> KB, the maximum size is <value> KB

[DVD+VR] specifies a maximum size for the VMGM_PGCI_UT of 20 Kbytes.

[DVD+VR] ERROR **6503** (ref. [DVD+VR] 3.3.2)
ERR_DVDVR_VMGI_TXTDT_MG_PRESENT
VMGI_MET: No TXTDT_MG is allowed, but <string value> specified <value>

The TXTDT_MG is not present on a DVD+RW Video disc, so the TXTDT_MG_SA in the VMGI_MAT must specify '0'.

- This requirement dates from before v1.0 of the DVD+RW Video specification and is no longer checked as of v1.0.

[DVD+VR] ERROR **6504** (ref. [DVD+VR] 3.3.2)
ERR_DVDVR_VMGM_TXTDT_MG_SIZE
The size of the VMGI TXTDT_MG is <value> KB; the maximum size is <value> KB.

The maximum size of the VMGI TXTDT_MG table is 1 sector or 2048 bytes.

- This is a new v1.0 DVD+RW Video spec requirement.

[DVD+VR] ERROR **6505** (ref. [DVD+VR] 3.3.2)
ERR_DVDVR_VMGI_MAX_CELLS
The number of cells in the VMGM_C_ADT (<value>) exceeds the number of cells (<value>) allowed in the VMGM_C_ADT.

The maximum number of Cells in the VMGM_C_ADT is 170.

[DVD+VR] ERROR **6507** (ref. DVD+VR 3.3.2.1)
ERR_DVDVR_VMGI_MAT_ADP_ID_ERR
VMGI_MAT (BP %d): ADP_ID has the value <value>b. It must be <value>b.

The ADP_ID in VMGI_MAT must be 1b.

[DVD+VR] ERROR **6510** (ref. [DVD+VR] 3.3.2)
ERR_DVDVR_VMGI_MAX_VOBUS
The number of VOBUs in the VMGM_VOBU_ADMAP (<value>) exceeds the maximum number of VOBUs (<value>) allowed in the VMGM_VOBU_ADMAP.

The maximum number of VOBUs in the VMGM_VOBU_ADMAP is 511.

[DVD+VR] ERROR **6515** (ref. [DVD+VR] 3.3.2.1)
ERR_DVDVR_VMG_CAT_REGION
VMG_CAT: All RMA fields must specify the value '0', indicating a multi-region DVD disc.

The recorded disc must be playable on DVD players of all regions.

[DVD+VR] ERROR **6516** (ref. [DVD+VR] 3.3.2.1)
ERR_DVDVR_VMGI_VTS_NS_LARGE
The VTS_Ns (<value>) is not in the valid range [1..<maximum value>].

The VTS_Ns must be in the range 1..3.

[DVD+VR] ERROR **6517** (ref. [DVD+VR] 3.3.2.1)
ERR_DVDVR_VMGI_FIRST8_PVR_ID_ILL
The first 8 bytes of the PVR_ID (‘<string value>’) must describe '<string value>'.

The first 8 bytes of the PVR_ID must contain the string “DVD+VR01”.

[DVD+VR] RECOMMENDATION VIOLATION **6520** (ref. [DVD+VR] 3.3.2.1)
ERR_DVDVR_VMGI_FP_PGCI_COMMAND
The FP_PGCI is recommended to contain a navcmd to link to the Title Menu but no such command is found.

The FP_PGCI is recommended to contain a link to the VMGM Title Menu, by means of a JumpSS navigation command, which should specify the Domain_ID ‘01b’ (for the VMGM domain) and Menu_ID ‘0010b’ (for the Title Menu). This message is reported as a Recommendation violation.

[DVD+VR] ERROR **6521** (ref. [DVD+VR] 3.2.8.1.)
ERR_DVDVR_VMGI_PLAYLIST_TITLE_ACCESS_ILL
All Play list Titles must be accessible through the Title Menu, but the Play list Title <value> is not accessible

All Play list Titles must be accessible via the Title menu. The PGC for the Title menu must specify a JumpTT for each of the Play list Titles defined in the TT_SRPT.

[DVD+VR] ERROR **6525** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_TT_SPR_NS_ILL
TT_SRPTI: The TT_SRP_Ns (<value>) must specify an even value.

The TT_SRPT consists of two sections of equal length, therefore the number of Title search pointers must be even.

[DVD+VR] ERROR **6528** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_TT_SRP_SA_ORDER_ILL
The start address of the first cell (<value>, C_IDN <value>, VOB_IDN <value>) of the <value> (<’Play list-’ | ’Full-’)Title search pointer is not sorted in the order of incrementing start addresses, because it is smaller than the start address of the first Cell of the <value> (<’Play list-’ | ’Full-’)Title (<value>, C_IDN <value>, VOB_IDN <value>).

The Titles on a DVD+RW Video disc must be sorted with ascending start addresses of the first cell used by the titles.

[DVD+VR] ERROR **6529** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_CHAPTERS_FULL_TITLES
TT_SRPT: The combined number of Chapters (part of titles) of all Full Titles is <value>, but may not exceed <value>.

The maximum number of Chapters (Part_of_Titles) in all Full Titles on the disc is 254.

[DVD+VR] ERROR **6530** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_TT_PB_TY_ILL_TT_TY
TT_SRP[<index>]: The TT_TY (<value>) must specify a One_Sequential_PGC_Title (<value>).

Only One-Sequential PGC titles are allowed on a DVD+RW Video disc.

[DVD+VR] ERROR **6531** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_TT_PB_TY_ILL_TT_PB_TY
TT_SRP[<index>]: The TT_PB_TY<id> (<value>) must specify <value>.

TT_PB_TY1 must specify '0': No Link/Jump/Call instruction as a Cell Command or Button Command in any Title.
TT_PB_TY2 must specify '1': All Titles contain Link/Jump/Call instructions as a Pre- or Post Command.
TT_PB_TY3 must specify '0': No Link/Jump/Call instruction as a Button Command in any Title.
TT_PB_TY4 must specify '1': All Titles contain a Link/Jump/Call instruction in the Title Domain.

[DVD+VR] ERROR **6532** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_TT_PB_TY_ILL_UOP
TT_SRP[<index>]: The UOP0 (<value>) must specify <value>, because Time_Play() and Time_Search() user operations are blocked for all Titles

The UOP0 must specify '1', which blocks Time_Play() and Time_Search() user operations for all Titles.

[DVD+VR] ERROR **6540** (ref. [DVD+VR] 3.3.2.3.1)
ERR_DVDVR_VMGM_LU_NS_ILL
VMGM_LU_Ns (<value>) may specify only <value> VMGM language units.

Only one language unit can be specified, so the VMGM_LU_Ns is maximum '1'.

[DVD+VR] ERROR **6541** (ref. [DVD+VR] 3.3.2.3.1)
ERR_DVDVR_VMGM_LU_SRP_MENU_ILL
VMGM_LU_SRP[<index>]: The VMGM_EXST (<value>) must indicate the existence of the Title Menu for this Language unit.

The VMGM_LU_SRP search pointer for the only allowed Language unit must indicate that the Entry PGC of the Title menu exists in this Language unit.

[DVD+VR] ERROR **6542** (ref. [DVD+VR] 3.3.2.3.1)
ERR_DVDVR_VMGM_LU_SRP_NS_ILL
The VMGM_LU[<index>] must contain at least <number> VMGM_PGCI_SRP pointers, but only <value> search pointers were found.

The VMGM language unit must contain at least 3 search pointers to PGCs

[DVD+VR] ERROR **6543** (ref. [DVD+VR] 3.3.2.3.2)

ERR_DVDVR_VMGM_LU_SRP1_ILL_DEST
VMGM_LU[<index>]: The VMGM_PGCI_SRP[<index>] must specify the Entry PGC for the Title Menu, but the Entry_type is <value> (‘<string value>’) and the Menu_ID is <value> (‘<string value>’).

The first VMGM_PGCI search pointer in the VMGM_LU must point to the Entry PGC for the Title menu. The Entry type field must specify ‘1’ and the Menu_ID must specify ‘0010b’ for the Title menu.

[DVD+VR] ERROR **6544** (ref. [DVD+VR] 3.3.2.3.2)
ERR_DVDVR_VMGM_LU_SRP23_ILL_DEST
VMGM_LU[<index>]: The VMGM_PGCI_SRP[<index>] must specify the Next Title <string value>PGC.

- Not implemented, because this is player-dependent and not distinguishable as the Next Title PGC or Next Title Extension PGC.

[DVD+VR] ERROR **6545** (ref. [DVD+VR] 3.3.2.3.2)
ERR_DVDVR_VMGM_LU_PGC_CAT_ILL
VMGM_LU[<index>]: The VMGM_PGCI_SRP[<index>] <string value> (<value>) must specify '0'.

For all VMGM_PGCI_SRPs the following fields must specify ‘0’:
- Entry type (except for the first VMGM_PGCI_SRP, which must specify the Entry PGC for the Title menu).
- Menu_ID (except for the first VMGM_PGCI_SRP, which must specify the Title menu).
- Block mode.
- Block type.
- PTL_ID_FLD.

[DVD+VR] ERROR **6546** (ref. [DVD+VR] 3.3.2.2)
ERR_DVDVR_VMGI_VCPS_TTN_AND_TTN
TT_SRPT: The number of video titles in the TT_SRPTI is <value> but should be equal to the number of video titles in the VCPS_TT_SRPTI (<value>)

Disabled or changed DVD VMGI Checks

None

9.4.5.2 DVD+RW Video Specific VTSI Checks

[DVD+VR] ERROR **6551** (ref. [DVD+VR] 3.3.3)
ERR_DVDVR_VTSI_NO_MENU
The VTSI may not contain a menu, <'VTSM_VOBS_SA' | 'VTSM_C_ADT_SA' | 'VTSM_VOBU_ADMAP_SA'> (<hexadecimal value>) must be 0x00000000.

The VTS Menu VOB is not allowed, so the VTSM_C_ADT does not exist and the VTSM_VOBU_ADMAP does not exist. Therefore, the VTSM_VOBS_SA, VTSM_C_ADT_SA and VTSM_VOBU_ADMAP_SA must specify '0'.

[DVD+VR] ERROR **6552** (ref. [DVD+VR] 3.3.3.1)
ERR_DVDVR_VTSI_VTSTT_VOBS_SA_ILL
The VTSTT_VOBS_SA (<hexadecimal value>) must be <hexadecimal value>. The VTS_TT_VOBS file starts at logical sector <hexadecimal value> and the VTSI file starts at logical sector <hexadecimal value>.

The VTSTT_VOBS_SA must point to the same sector in all VTSIs on the disc. This means that the value of the VTSTT_VOBS_SA added to the start of the VTSI must be the same in all VTSIs.

[DVD+VR] ERROR **6553** (ref. [DVD+VR] 3.3.3, Table 3-4)
ERR_DVDVR_VTSI_MAT_VOBU_ADMAP_SA_INVALID
The VTS_VOBU_ADMAP_SA field is <value>, but must be in range [7..32].

[DVD+VR] ERROR **6554** (ref. [DVD+VR] 3.2.1)
ERR_DVDVR_VTSI_VID_COMPR_MPEG1_ILL
VTS_V_ATR: The Video compression mode (<value> '<string value>') cannot be used with the specified resolution <value> '<string value>'

[DVD+VR] allows for an MPEG1 Video compression mode ('00b') only if the picture resolution used is 352x240 (525/60 system) or 352x288 (625/50 system).

[DVD+VR] ERROR **6555** (ref. [DVD+VR] 3.3.3.1.1)
ERR_DVDVR_VTSI_VIDPAR_ALREADY_USED
The VTSI must specify unique video parameters. The specified video parameters (<value>x<value> '<string value>') are already used in VTSI #<value>.

Of the 3 VTSs that are allowed on a DVD+RW Video disc, no VTS may use the same video parameters as any other VTS on the disc.

[DVD+VR] ERROR **6556** (ref. [DVD+VR] 3.3.3.1.1)
ERR_DVDVR_VTSI_ASPECT_RATIO_ILL
VTS_V_ATR: The Aspect_ratio (<value> '<string value>') must specify <value> '<string value>'.

The VTSI must specify '00b', (4:3) as the aspect ratio. The aspect ratio can be overruled by the real-time attributes in the PCI_GI.

[DVD+VR] ERROR **6557** (ref. [DVD+VR] 3.3.3.1.1)
ERR_DVDVR_VTSI_LINE21_ILL
The Line21_switch_<id> (<value>) must be <value> in the case of <'PAL' | 'NTSC'> TV system.

The Line21_switch_1 and Line21_switch_2 must be set to '1b' in an NTSC (525/60) stream and '0b' in a PAL (625/50) stream.

[DVD+VR] ERROR **6558** (ref. [DVD+VR] 3.3.3.1.1)
ERR_DVDVR_VTSI_LETTERBOX_ILL
VTS_V_ATR: The Source picture letterboxed field (<value> '<string value>') must specify (<value> '<string value>').

The Source picture letterboxed must indicate '0b', (not letterboxed). The Source picture letterboxed can be overruled by the real-time attributes in the PCI_GI.

[DVD+VR] ERROR **6559** (ref. [DVD+VR] 3.3.3.1.1)
ERR_DVDVR_VTSI_SOURCE_PIC_RES_ILL
VTS_V_ATR: The Source picture resolution field (<value> '<string value>') is not allowed.

The Source picture letterboxed must indicate one of these resolutions:

NTSC (525/60)	**PAL** (625/50)
720x480	720x576
352x480	352x576
352x240	352x288

So compared to DVD, the 704x480 or 704x576 Source picture resolution is not allowed.

[DVD+VR] ERROR **6560** (ref. [DVD+VR] 3.3.3.1.1)
ERR_DVDVR_VTSI_CAMERA_MODE_ILL
VTS_V_ATR: The Film camera mode field (<value> '<string value>') must specify (<value> '<string value>').

The Film camera mode must indicate 'camera mode'. The Film camera mode can be overruled by the real-time attributes in the PCI_GI.

[DVD+VR] ERROR **6561** (ref. [DVD+VR] 3.3.3.1.2)
ERR_DVDVR_VTSI_AUDIO_SET_NOT_DIFF
VTS_AST_Ns: <value> audio sets are defined, some audio sets define the same audio type (<value> '<string value>').

The VTS_AST_Ns defines the number of different audio stream attribute sets. So no audio attributes in the VTS_AST_ATR table must specify identical attributes.

[DVD+VR] ERROR **6562** (ref. [DVD+VR] 3.3.3.1.3)
ERR_DVDVR_VTSI_DEF_AUDIO_SET_NOT_DEFINED
VTS_AST_ATRT: <value> audio sets are defined, at least 2ch Dolby AC-3 and 2ch MPEG1 audio must be defined. No <string value> audio set is defined.

When the VTS_AST_ATRT table specifies 7 or 8 audio sets, at least 2 channel Dolby AC-3 and 2 channel MPEG-1 audio sets must de defined.

[DVD+VR] ERROR **6563** (ref. [DVD+VR] 3.3.3.1.3)
ERR_DVDVR_VTSI_AUDIO_TYPE_ILL
VTS_AST_ATRT: The Audio type (<value> '<string value>') must specify <value> '<string value>'.

The Audio type field must specify '00b', 'not specified'.

[DVD+VR] ERROR **6564** (ref. [DVD+VR] 3.3.3.1.3)
ERR_DVDVR_VTSI_AUDIO_APPL_MODE_ILL
VTS_AST_ATRT: The Audio application mode (<value> '<string value>') must specify <value> '<string value>'.

The Audio application mode field must specify '10b', 'Surround mode'.

[DVD+VR] ERROR **6565** (ref. [DVD+VR] 3.3.3.1.4)
ERR_DVDVR_VTSI_SUBPIC_NS
VTS_SPST_Ns: Only <value> Sub-picture streams are allowed, but <value> Sub-picture streams are defined.

[DVD+VR] only allows for 1 Sub-picture stream.

[DVD+VR] ERROR **6566** (ref. [DVD+VR] 3.3.3.1.5)
ERR_DVDVR_VTSI_SUBPIC_NOT_ZERO
VTS_SPST_ATRT[<index>]: All fields must be <value>.

All fields in the VTS_SPST_ATRT must specify '0'.

[DVD+VR] ERROR **6567** (ref. [DVD+VR] 3.2.1)
ERR_DVDVR_VTSI_TVSYS_ILL
VTS_V_ATR: The TV system (<value> '<string value>') must be equal to the TV system from the VMGI (<value> '<string value>').

The TV system video attributes in VMGI and VTSI shall be identical for all VTSs on a disc.

[DVD+VR] ERROR **6568** (ref. [DVD+VR] 3.3.3, Table 3-4)
ERR_DVDVR_VTSI_VTS_TMAPT_MISSING
VTSI_MAT: The VTS_TMAPT_SA is 0. The VTS_TMAPs must be present for all Titles.

[DVD+VR] ERROR **6569** (ref. [DVD+VR] 3.3.3, Table 3-4)
ERR_DVDVR_VTSI_VTSM_PGCI_UT_MISSING
VTSI_MAT: The VTSM_PGCI_UT_SA is 0. The VTSM_PGCI_UT must be present.

[DVD+VR] ERROR **6571** (ref. [DVD+VR] 3.3.3.2)
ERR_DVDVR_VTSI_PTT_PGCN_ERR
All PGCN for the same TTU must specify the same value (<value>).

Since all Titles are One_Sequential_PGC_Titles, all Program Chain numbers (PGCN) within the same TTU must be identical.

[DVD+VR] ERROR **6572** (ref. [DVD+VR] 3.3.3.3)
ERR_DVDVR_VTSI_VTS_PGCI_SRP_NUM
The number of VTS_PGCI_SRP (<value>) must be equal to the number of Title Units (<value>) in this VTSI.

The number of VTS_PGCI search pointers must be equal to the number of Title units in this VTSI.

[DVD+VR] ERROR **6573** (ref. [DVD+VR] 3.3.3.3)
ERR_DVDVR_VTSI_VTS_PGCI_SRP_ORDER
The VTS_PGCI_SRP must be recorded in the same order as the TTUs.

The VTS_PGCI search pointers must be recorded in the same order as the Title Units. This means that the VTS_TTN in the VTS_PGCI_SRP. VTS_PGC_CAT must be assigned in ascending and consecutive order.

[DVD+VR] ERROR **6574** (ref. [DVD+VR] 3.3.3.3)
ERR_DVDVR_VTSI_VTS_PGCI_SA_USED
VTS_PGCIT: All VTS_PGCI_SA values should be different, but VTS_PGCI_SA[<index>] equals VTS_PGCI_SA[<index>] (value <value>).

The VTS_PGCI_SRP must specify a VTS_PGCI_SA that is different from all the other VTS_PGCI_SRPs.

[DVD+VR] ERROR **6575** (ref. [DVD+VR] 3.3.3.3)
ERR_DVDVR_VTSI_PGC_NON_ENTRY
VTS_PGCI_SRP[<index>]: The PGC must be an Entry PGC, but Entry type is <value> '<string value>'.

All PGCs must be Entry PGCs.

[DVD+VR] ERROR **6576** (ref. [DVD+VR] 3.3.3.3)
ERR_DVDVR_VTSI_PGC_NON_ZERO_BIT
VTS_PGCI_SRP[<index>]: The PGC must specify all zero bits for <string value>.

All PGC Search pointers must specify '0' for the Block mode, Block type and PTL_ID_FLD fields.

[DVD+VR] ERROR **6580** (ref. [DVD+VR] 3.3.3.4)
ERR_DVDVR_VTSI_VTSM_NS_ILL
VTSM_PGCI_UTI: The VTSM_LU_Ns (<value>) must be <value>.

Exactly 1 VTSM_LU must be specified.

[DVD+VR] ERROR **6581** (ref. [DVD+VR] 3.3.3.4)
ERR_DVDVR_VTSI_VTSM_EXST_ILL
VTSM_LU_SRP[<index>]: The VTSM_EXST (<value>) must specify <value> '<string value>'

The VTSM_EXST of the only VTSM_LU_SRP must specify that only the Root Menu exists; it must contain the value '1000 0000b'.

[DVD+VR] ERROR **6582** (ref. [DVD+VR] 3.3.3.4)
ERR_DVDVR_VTSI_VTSM_PGCI_SRP_LARGE
VTSM_LUI: <value> PGC search pointers specified, only <value> allowed.

The VTSM language unit shall contain exactly one VTSM_PGCI_SRP.

[DVD+VR] ERROR **6583** (ref. [DVD+VR] 3.3.3.4)
ERR_DVDVR_VTSI_VTSM_PGC_DEST_ILL
VTSM_LU[<index>]: The VTSM_PGCI_SRP[<index>] must specify the Entry PGC for the Root Menu, but the Entry type field is <value> ('<string value>') and the Menu_ID is <value> ('<string value>').

The VTSM_PGC must specify '8300 0000h', indicating that the PGC is the Entry PGC for the Root menu.

[DVD+VR] ERROR **6584** (ref. [DVD+VR] 3.3.3.4)
ERR_DVDVR_VTSI_VTSM_PGC_CAT_ILL
VTSM_LU[<index>]: The VTSM_PGCI_SRP[<index>] <string value> must specify '0'.

The Block mode, Block type and PTL_ID_FLD fields of the VTSM_PGCI_SRP must be '0'.

[DVD+VR] ERROR **6585** (ref. [DVD+VR] N/A)
ERR_DVDVR_VTSI_VTSM_PGC_CELLS_ILL
VTSM_PGC[<index>]: The PGC cannot specify any Cells or Programs (dummy PGC).

The VTSM_PGC cannot specify Cells or Programs, because the VTSM_VOB is not allowed, thus a dummy PGC must be used.

[DVD+VR] ERROR **6586** (ref. [DVD+VR] 3.3.3.5)
ERR_DVDVR_VTSI_VTS_TMU_ILL
VTS_TMAPT: The TMU (<value>) shall specify <value>.

The TMU for the VTS_TMAP must specify ‘00h’: the VTS may not contain any map entries. The VTS_TMAP must be present, but with no map entries for [DVD+VR].

[DVD+VR] ERROR **6587** (ref. [DVD+VR] 3.3.3.5)
ERR_DVDVR_VTSI_VTS_TMAP_MAP_EN_NS_ILL
VTS_TMAPT: MAP_EN_Ns (<value>) must be <value>.

The MAP_EN_Ns in the VTS_TMAP must specify ‘0’: the Title may not contain any time map entries.

[DVD+VR] ERROR **6590** (ref. [DVD+VR] 3.3.3.6)
ERR_DVDVR_VTSI_VTS_VOB_NS_ILL
VTS_C_ADT: The VTS_VOB_Ns (<value>) must be <value>.

The VTS_VOB_Ns must contain the value ‘1’, because all VOBs on a DVD+RW Video disc have the same VOB number (1).

[DVD+VR] ERROR **6591** (ref. [DVD+VR] 3.3.3.6)
ERR_DVDVR_VTSI_VTS_CPI_VOBIDN_ILL
VTS_C_ADT.VTS_CPI[<index>]: The VTS_VOB_IDN (<value>) must specify the value <value>.

All VTS_CPI entries must specify the same value (‘1’) for the VTS_VOB_IDN, because all VOBs on a DVD+RW Video disc have the same VOB number (1).

[DVD+VR] ERROR **6592** (ref. [DVD+VR] 3.3.3.6)
ERR_DVDVR_VTSI_VTS_FIRST_CPI_CIDN_ILL
VTS_C_ADT.VTS_CPI[<index>]: The first VTS_C_IDN (<value>) must be ‘1’.

The VTS_CPI VTS_C_IDN values must start from 1.

[DVD+VR] ERROR **6593** (ref. [DVD+VR] 3.3.3.6)
ERR_DVDVR_VTSI_VTS_CPI_CIDN_ILL
VTS_C_ADT.VTS_CPI[<index>]: The VTS_C_IDN (<value>) must be equal to the previous VTS_CPI.VTS_C_IDN (<value>) increased by 1.

The VTS_CPI VTS_C_IDN values must start from 1 and increment by one up to and including 254.

[DVD+VR] ERROR **6594** (ref. [DVD+VR] 3.3.3.6)
ERR_DVDVR_VTSI_VTS_CPI_LOW
VTS_C_ADT: The number of recorded VTS_CPI (<value>) must be 254.

The VTS_C_ADT must contain exactly 254 VTS_CPI blocks.

[DVD+VR] ERROR **6595** (ref. [DVD+VR] 3.3.3.6)
ERR_DVDVR_VTSI_VTS_CP_USED_IN_PREV_VTS
VTS_C_ADT.VTS_CPI[<index>]: The VTS_CP_SA (<value>) and VTS_CP_EA (<value>) both must specify '0' because the Cell Piece with C_IDN <value> and VOB_IDN <value> is used in VTS <value>.

It is not allowed to use the same Cell ID number in more than one VTS. The VTS_C_ADT table must specify '0' for both the Cell start address and Cell end address if this cell is being used by another VTS. For VTS 1, this check is not performed. For VTS 2, the check is performed to see if cells are specified that are used in VTS 1. For VTS 3, the check is performed to see if cells are specified that are used in VTS 1 or VTS 2.

[DVD+VR] ERROR **6598** (ref. [DVD+VR] 3.3.3.7)
ERR_DVDVR_VTSI_VOBU_ADMAP_IDENTICAL
VTS_VOBU_ADMAP: different for VTS <value> and VTS <value>.

The complete VTS_VOBU_ADMAP table must be identical for all VTSs in the DVD-Video zone.

[DVD+VR] ERROR **6599** (ref. [DVD+VR] 3.3.3 - Table 3.4)
ERR_DVDVR_VTSI_VOBU_ADMAP_LARGE
VTS_VOBU_ADMAP: The number of VOBUs (<value>) is too large, maximum <value> VOBUs allowed.

The VTSI can specify a maximum of 40959 VOBUs.

Disabled or changed DVD VTSI Checks

[DVD+VR] ERROR **4338** (ref. DVD-3 Table 4.2.8-2)
ERR_DVD_VTS_CP_SA_LOWER_ILL
VTS_CPI[<index>] : VTS_CP_SA (<value>) cannot be lower than the previous VTS_CP_SA (<value>) within a VOB

This check is only disabled for the VTS_CP_ADT, but since DVD+RW Video does not allow a VTSM, this check is effectively disabled completely.

[DVD+VR] ERROR **4344** (ref. DVD-3 Table 4.2.8-2)
ERR_DVD_VTS_CP_EA_LOWER
VTS_CPI[<index>] : VTS_CP_EA (<value>) must be larger than VTS_CP_SA (<value>)

The end address of the Cell Piece must be larger than the start address of the Cell Piece. This check needs to be changed for DVD+VR, because the Cell Piece table contains references to all 254 possible Cell Pieces, not just the Cell Pieces belonging to the VTS like in the DVD spec. This means that Cell Piece addresses may be 0. This check is only performed for DVD+RW Video if both the VTS_CP_SA and VTS_CP_EA are not equal to 0x0.

9.4.5.3 DVD+RW Video Specific PGCI Checks

All the PGC constraints listed here only apply to Program Chains for a Title, unless explicitly stated otherwise.

[DVD+VR] ERROR **6601** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_PGCI_AST_AVAIL
PGC_AST_CTLT: <value> audio streams are specified as available. At most two audio streams should be available.

A PGC may specify zero, one or two audio streams.

[DVD+VR] ERROR **6602** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_PGCI_SPST_AVAIL
PGC_SPST_CTLT: No Sub-picture streams are specified as available. One Sub-picture stream should be available.

The Availability flag of one Sub-picture stream in the PGC_SPST_CTLT must be set to ‘1b’, indicating an available Sub-picture stream, meaning that exactly one Sub Picture is available.

[DVD+VR] ERROR **6603** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_PGCI_SPST_AVAIL2
PGC_SPST_CTLT: Only the first PGC_SPST_CTL shall be specified as available.

Only the Availability flag of the first Sub-picture stream in the PGC_SPST_CTLT may be set to ‘1b’.

[DVD+VR] ERROR **6604** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_PGCI_SPST_NOT_NULL
PGC_SPST_CTL[<index>]: All bits must be ‘0’.

All bits of the PGC_SPST_CTLT must be ‘0b’, except for the first PGC_SPST_CLT.

[DVD+VR] ERROR **6605** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_PGCI_NV_CTL_PG_PLAY
PGC_NV_CTL: The PG Playback mode (<value> ‘<string value>’) must be set to <value> ‘<string value>’.

The PG Playback mode in PGC_NV_CTL must be set to ‘00000000b’, indicating sequential playback.

[DVD+VR] ERROR **6606** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_PGCI_NV_CTL_STILL_TIME
PGC_NV_CTL: The Still time value (<value> ‘<string value>’) must be set to <value> ‘<string value>’.

The Still time value in PGC_NV_CTL must be set to ‘00000000b’, indicating no still.

[DVD+VR] ERROR **6607** (ref. [DVD+VR] 3.2.8.1)
ERR_DVDVR_PGCI_UOP_NOTBLOCKED
PGC_UOP_CTL: Menu_ID <value> (<string value>) demands that for this PGC UOP[10..16] be blocked (set to '1'). However, UOP[<string value>] are permitted.

The UOP10…UOP16 fields in the PGC for a Root menu must be set to ‘1’ in order to block the menu operations. This check is performed on PGCs that belong to the VMGM domain. The PGCs for the VMGM domain form the Root menu and cannot specify any cells, therefor these UOP operations cannot be enabled.

[DVD+VR] ERROR **6608** (ref. [DVD+VR] 3.2.8.1)
ERR_DVDVR_PGCI_UOP_NOTBLOCKED2
PGC_UOP_CTL: There are no real titles on the disc, but UOP[<string value>] are still permitted. UOP[10..16] must be blocked (set to '1') when no real titles are available

When there are no real titles on the disc, the UOP10…UOP16 fields in the PGC for a Root menu must be set to '1' in order to block the menu operations.

[DVD+VR] ERROR **6609** (ref. [DVD+VR] 3.2.8.1)
ERR_DVDVR_PGCI_UOP_NOTBLOCKED3
PGC_UOP_CTL: PGCI <value> of the VMGM must have the UOP[10..16] blocked but this PGC specifies UOP[<string value>] as permitted

Only the first PGC in the VMGM may have the UOP10…UOP16 fields in the PGC_OUP_CTL set to '0' (permitted); all other PGCs in the VMGM must have these operations blocked.

[DVD+VR] ERROR **6610** (ref. [DVD+VR] 3.3.4.2)
ERR_DVDVR_PGCI_COMMAND_NRS_ILL
PGC_CMDT: The number of Navigation commands (<value>) must be <value>.

The PGC Command Table must contain exactly 3 commands.

[DVD+VR] ERROR **6612** (ref. [DVD+VR] 3.3.4.2)
ERR_DVDVR_PGCI_ILL_POST_CMD
PGC_CMDT: A PGC must contain at least 1 POST command

At least one of the 3 allowed navigation commands shall be a Post command.

[DVD+VR] ERROR **6613** (ref. [DVD+VR] 3.3.4.2)
ERR_DVDVR_PGCI_ILL_POST_CMD2
PGC_CMDT: The last Post-command (<string value>) must be <string value>

The last Post command of the PGC must be a CallSS to the PGCN#2 of the VMGM.

[DVD+VR] ERROR **6620** (ref. [DVD+VR] 3.3.4.3)
ERR_DVDVR_PGCI_ILL_CELL_ATTRIBUTE
C_PBI[<index>]: The <string value> field must specify <value> '<string value>'.

The C_PBIT cannot specify a Cell that is part of an Angle block or a Cell that exists in an Interleaved block. This means that in C_CAT:

- Cell Block type and Cell Block mode must both be set to '0'.
- Seamless Angle Change flag must be set to '0'.
- Interleaved allocation flag must be set to '0'.
- STC_discontinuity flag must be set to '1' for first Cell of a PGCI.

[DVD+VR] ERROR **6622** (ref. [DVD+VR] 3.3.4.3)
ERR_DVDVR_PGCI_ILL_SEAMLESS
C_PBI[<index>]: The Seamless playback flag (<value>) must be set to '0' when the previous cell is part of a different VOB as the current Cell

Not implemented yet, because it is a cross check, or can not be checked at all if a VOB boundary can not be detected.

[DVD+VR] ERROR **6630** (ref. [DVD+VR] 3.3.4.4)
ERR_DVDVR_PGCI_C_POSIT_VOBIDN_ILL
C_POSI[<index>]: The VOB_IDN (<value>) must specify <value>.

The VOB_IDN of all Cells in the PGC must contain the value '1'.

[DVD+VR] ERROR **6630** (ref. [DVD+VR] 3.3.4.4)
ERR_DVDVR_PGCI_C_POSIT_VOBIDN_ILL
C_POSI[<index>]: The VOB_IDN <value> must specify <value>.

Since all VOBs have an identical VOB_IDN of 1 in DVD+RW Video, all Cells in the PGC must specify a VOB_IDN of 1 too.

[DVD+VR] ERROR **6635** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGC_FREESPACE_ADDRILL
PGC_GI: The <string value> (<hexadecimal value>) must specify <hexadecimal value> in a Free Space Title PGC

This error reports that one of the following address fields in the PGC_GI table does not specify the correct value for a Free Space Title PGC:

PGC_GI field	Mandatory value in a Free Space Title PGC
PGC_CMDT_SA	00ECh
PGC_PGMAP_SA	010Ch
C_PBIT_SA	010Eh
C_POSIT_SA	0126h

[DVD+VR] ERROR **6636** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGC_FREESPACE_PGCILL
PGC_GI: A Free Space Title PGC must specify 1 program and 1 cell, while <value> programs and <value> cells specified

A Free Space Title PGC must specify exactly 1 program and 1 cell.

[DVD+VR] ERROR **6640** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE
<string value>: the <string value> field (<value>) must specify <value> for a Free Space Title PGC

This error reports that one of the following fields does not specify the correct value for a Free Space Title PGC:

PGC table	field	Mandatory value in a Free Space Title PGC
PGC_PB_TM	Hour (ten's)	0
C_PBTM of the C_PBI	Hour (units)	0
	Minute (ten's)	0
	Minute (units)	0
	Second (ten's)	0
	Second (units)	1
	tc_flag	Conform to the TV system
	Video frame (ten's)	0
	Video frame (units)	0

C_CAT of the C_PBIT	Cell Block mode	0
	Cell Block type	0
	Seamless playback flag	0
	Interleaved allocation flag	0
	STC discontinuity flag	1
	Seamless Angle Change flag	0
	Cell playback mode	0
	Access Restriction flag	1
	Cell type	0
	Cell Still time	0
	Cell Command number	0
C_POSI	C_IDN	255

[DVD+VR] ERROR **6641** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_UOP_FREESPACE_NOTBLOCKED
PGC_UOP_CTL: Free Space Titles PGCs must have UOP[10..16] blocked, whereas UOP[<string value>] are permitted in this Free Space Title PGC

The UOP10..16 fields in the PGC_UOP_CTL table in a Free Space Title PGC must be blocked.

[DVD+VR] ERROR **6642** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE_ILL_ATTRIB
<string value>[<value>]: <string value> first attribute (<hexadecimal value>) must specify <hexadecimal value> in a Free Space Title PGC

The first Audio or Subpicture must be available and must specify ‘0’ for any decoding stream number. All other Audio and Subpicture streams must be unavailable.

[DVD+VR] ERROR **6643** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE_PGC_NV_CTL
All fields of the PGC_NV_CTL in a Free Space Title PGC must be '0' but <string value> specified <value>

This error reports that one of the following fields of the PGC_NV_CTL does not specify the correct value for a Free Space Title PGC:

Field of the PGC_NV_CTL	Mandatory value in a Free Space Title PGC
Next_PGCN	0
Previous_PGCN	0
GoUp_PGCN	0
PG Playback mode	0
Still time value	0

[DVD+VR] ERROR **6644** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE_PGC_SP_PLT
The <string value> field of the PGC_SP_PLT[<value>] (<value>) shall specify <value> for a Free Space Title PGC

This error reports that one of the following fields of each of the 16 PGC_SP_PLT entries does not specify the correct value for a Free Space Title PGC:

Field of the PGC_SP_PLT	Mandatory value in a Free Space Title PGC
Luminance signal Y	127
Color difference signal Cr	127
Color difference signal Cb	127

[DVD+VR] ERROR **6645** (ref. [DVD+VR] 3.3.5)
ERR_DVDVR_PGCI_FREESPACE_LVOBU_EA
C_PBI[0]: the C_LVOBU_EA <value> must be larger then C_LVOBU_SA <value>.

[DVD+VR] ERROR **6650** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE_CMD
The number of <string value>-commands (<value>) in a Free Space Title PGC must be <value>

The number of commands in a Free Space Title PGC must comply to:

Type of command	Mandatory number of commands in a Free Space Title PGC
Pre command	2
Post command	1
Cell command	0

[DVD+VR] ERROR **6651** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE_CMD_EA
The end address of the PGC_CMDT (<value>) must be <value>

The end address of the PGC_CMDT in a Free Space Title must be 31 (1Fh), due to the mandatory number of Pre-, Post- and Cell commands.

[DVD+VR] ERROR **6652** (ref. [DVD+VR] 3.3.5)
ERR_DVDVR_PGCI_FREESPACE_CMD_PGC2_VMGM
The second PRE command '<name>' must be '<name>'.

The first Pre command in a Free Space Title PGC can specify any command, but the second (and last) Pre command in a Free Space Title PGC must be a CallSS to the PGCN #2 of the VMGM domain.

[DVD+VR] ERROR **6653** (ref. [DVD+VR] 3.3.5.)
ERR_DVDVR_PGCI_FREESPACE_PGMAP
PGC_PGMAP: The EN_CN (<value>) for the only Cell in the Program must be <value>

The EN_CN of the only Cell in the Free Space Title PGC must specify the value 255.

[DVD+VR] ERROR **6655** (ref. [DVD+VR] 3.3.3.4.)
ERR_DVDVR_PGCI_ROOT_MENU_ILL
PGC_GI: The PGC for the Root Menu must point to a dummy PGC, the number of programs (<value>) and number of Cells (<value>) must both be '0'

The PGC for the Root menu in the VTSM domain must point to a dummy PGC, which does not specify any programs or cells. This check is only performed on PGCs from the VMGM domain, which form the Root menu.

[DVD+VR] ERROR **6656** (ref. [DVD+VR] 3.3.3.4.)
ERR_DVDVR_PGCI_CMD_ROOT_MENU
The Root menu PGC did not specify a JumpSS to the Title menu

The PGC for the Root menu must specify a JumpSS to the Title menu as one of its navigation commands. This check is only performed on PGCs from the VTSM domain.

Disabled or changed DVD PGCI Checks

None

9.4.5.4 DVD+RW Video Specific Navigation Command Checks

[DVD+VR] ERROR **6671** (ref. [DVD+VR] 3.3.2.3.2)
ERR_DVDVR_NAVCMD_ILL_COUNTER_MODE
<navigation command name>: GPRM<number> may not be used in Counter mode.

General Parameter 0 … 5 are not allowed to be used in Counter mode. This prevents the navigation command 'SetGPRMMD' from specifying the GPRM0 … GPRM5 in b19..b16 (GPRMN Mode change) when b23 (mode) equals '1b' (Counter mode).

[DVD+VR] ERROR **6672** (ref. [DVD+VR] 3.3.2.3.2)
ERR_DVDVR_NAVCMD_GPRM_ILL_CMD_TYPE
<navigation command name>: The value of GPRM<number> may not be changed in a <'Button' | 'Cell'> Command.

The value of the General Parameter 0 … 5 are not allowed to be changed in a Cell- or Button command. Therefor, these commands are not allowed to use these GPRMs:

Command class	Specific command	#n
Set	Set GPRM#n = immediate value	0…5
	Set SetSystem Swap GPRM#n <> SPRM#m	0…5
Compare	Set GPRM#n = immediate value	0…5
	Set SetSystem Swap GPRM#n <> SPRM#m	0…5
Set Link	Set GPRM#n = immediate value	0…5
	Set SetSystem Swap GPRM#n <> SPRM#m	0…5
Set Compare Link	Set GPRM#n = immediate value	0…5
	Set SetSystem Swap GPRM#n <> SPRM#m	0…5
Compare and Set Link	Set GPRM#n = immediate value	0…5
	Set SetSystem Swap GPRM#n <> SPRM#m	0…5
Compare Set and Link	Set GPRM#n = immediate value	0…5
	Set SetSystem Swap GPRM#n <> SPRM#m	0…5
SetSystem	Set GPRMMD GPRM#n	0…5

Disabled or changed DVD Navigation Command Checks

None

9.4.5.5 DVD+RW Video Specific PCI Checks

These checks relate to the PCI data structure.

The PCI_GI has 32 reserved bytes that are re-defined for DVD+RW Video. The following are the checks related to the PCI_GI Extension data for DVD+RW Video.

9.4.5.5.1 PCI_GI (Extension) Checks

[DVD+VR] INFORMATION **6701** (ref. [DVD+VR] 3.4.1)
ERR_DVDVR_PCI_GI_EXT_NOT_DEFINED
The reserved last 32 bytes of PCI_GI Extension don't have the DVD+RW Video specific information encoded.

This is only reported as an **information** message, by the debug version of the verifier.

[DVD+VR] ERROR **6702** (ref. [DVD+VR] 3.4.1 / DVD-3 2.1)
ERR_DVDVR_PCI_GI_EXT_RESERVED_ ILL
PCI_GI Ext: The reserved field bits must all be zero.

This message is reported for any of the reserved fields in the PCI_GI Extension.

[DVD+VR] ERROR **6703** (ref. [DVD+VR] 3.4.1(8))
ERR_DVDVR_PCI_GI_XI_RTA_VAL_RESERVED
PCI_GI_XI: The 'RTA Validity' field has the reserved value 1.The allowed values are 0,2 and 3.

[DVD+VR] ERROR **6704** (ref. [DVD+VR] 3.4.1(8))
ERR_DVDVR_PCI_GI_XI_REC_INFO_RESERVED
PCI_GI_XI: The 'REC Info Validity' field has the reserved value 1. The allowed values are 0, 2 and 3.

[DVD+VR] ERROR **6705** (ref. [DVD+VR] 3.4.1(9))
ERR_DVDVR_PCI_GI_CHNG_FLD_ILL
The CHNG_FLD (<value>) must be greater than zero if 'RTA Validity' or 'Rec Info Validity' (or both) contain the value 3.

The CHNG_FLD can have only positive integer values. When the PCI_GI_XI's RTA Validity field has value 3 or Rec Info Validity has value 3 or both the fields have the same value, then, the CHNG_FLD must be greater than zero.

[DVD+VR] ERROR **6706** (ref. [DVD+VR] 3.4.1(10))
ERR_DVDVR_PCI_GI_RT_ATR_ILL
RT_ATR_<1|2>: <Aspect_ratio | Film bit | Subtitling mode | Surround Type | Reserved1 | Reserved2 | Reserved3> has <value>but must be 0 when 'RTA Validity' is <value>.

When the PCI_GI_XI's RTA Validity field has value 0, the RT_ATR_1 bytes must all be zero and when the RTA Validity field has value '0 or '2, RT_ATR_2 must be all zero.

[DVD+VR] ERROR **6707** (ref. [DVD+VR] 3.4.1(10/11))
ERR_DVDVR_PCI_GI_RT_ATR_ASPECT_RATIO_ILL
RT_ATR_<1|2>: The 'aspect ratio' field has an illegal value <value>. Allowed values are 1,2,4,7,8,11,13 and 14.

Allowed values are 8, 1, 2,11,4, 13,14 and 7.

[DVD+VR] ERROR **6708** (ref. [DVD+VR] 3.4.1(10/11))
ERR_DVDVR_PCI_GI_RT_ATR_SUBTIT_MODE_RESERVED
RT_ATR_<1|2>: The 'subtitling mode' field has reserved value <value>. Allowed values are 0,1 and 2.

The allowed values are '00b','01b' & '10b'.

[DVD+VR] ERROR **6709** (ref. [DVD+VR] 3.4.1(10/11))
ERR_DVDVR_PCI_GI_RT_ATR_SURROUND_TYPE_RESERVED
RT_ATR_<1|2>: The 'surround type' field has reserved value <value>. Allowed values are 0 and 2.

The allowed values are 0 and 2.

[DVD+VR] ERROR **6710** (ref. [DVD+VR] 3.4.1(12/14))
ERR_DVDVR_PCI_GI_REC_DATE_ILL
REC_DATE_<1|2>: '<Year|Week|Month|Day|DS|TM|Timezone|TZ_sign>' field has <value>, but must be 0 when 'REC Info Validity' is <value>.

When PCI_GI_XI's REC Info Validity has value '00b', the REC_DATE_1 bytes must be zero and when REC Info Validity is '00b' or '01b', REC_DATE_2 must be 'zero'.

[DVD+VR] ERROR **6711** (ref. [DVD+VR] 3.4.1(12/14))
ERR_DVDVR_PCI_GI_REC_DATE_RESERVED
REC_DATE_<1|2>: '<Year|Month|Day|time zone>' has reserved value <value>.

The allowed values for the Year field are: 00-99 & 'FFh'.
The allowed values for the Month field are: 01-12 & '1Fh'.
The allowed values for the Day field are: 01-31 & '3Fh'

[DVD+VR] ERROR **6712** (ref. [DVD+VR] 3.4.1(13/15))
ERR_DVDVR_PCI_GI_REC_TIME_ILL
REC_TIME_<1|2>: '<Hour|minute|Second|Video Frame>' field has <value> but must be 0 if 'Rec Info Validity' is '<value>'.

When PCI_GI_XI's REC Info Validity has the value '00b', the REC_TIME fields must be all zero.

[DVD+VR] ERROR **6713** (ref. [DVD+VR] 3.4.1(13/15))
ERR_DVDVR_PCI_GI_REC_TIME_ RESERVED
REC_TIME_<1|2>: '<Hour|Minute|Second >' field has reserved value <value>.

The allowed values for the Hour field are '00'-'23' and 'FF'.
The allowed values for the Minute are '00'-'59' and 'FF'.
The allowed values for the Second are '00'-'59' and 'FF'.

[DVD+VR] ERROR **6714** (ref. [DVD+VR] 3.4.1(12/14))
ERR_DVDVR_PCI_GI_ILL_BCD_VALUE
REC_DATE_<1|2>: Year (<tens | units>) has <value> which is an illegal BCD value.

[DVD+VR] ERROR **6715** (ref. [DVD+VR] 3.4.1(13/15))
ERR_DVDVR_PCI_GI_REC_TIME_VIDEO_FRAME_RESERVED
REC_TIME_<1|2' Video Frame' field has a reserved value <value>.

The allowed values are '00'-'29' and 'FF' if TV system is NTSC, and '00'-'24' or 'FF' if TV system is PAL.

[DVD+VR] ODDITY **6716** (ref. [DVD+VR] 3.4.1(12/14)
ERR_DVDVR_PCI_GI_REC_DATE_INVALID
REC_DATE_<1 | 2>: The field '<field name>' has an odd value <value>.

The recording date cannot be earlier than 2000, since this is about the date that the first data has been generated. This is only reported as an oddity, since it is not required but may indicate a mistake in the coding.

Note: This check has not been implemented.

[DVD+VR] ERROR **6717** (ref. [DVD+VR] 3.4.1(12/13/14/15))
ERR_DVDVR_PCI_GI_RANGE
<REC_DATE | REC_TIME>_<1 | 2>: Field '<field name>' has an illegal value <value>. It must be in the range [<low value>..<high value>].

The value encoded for the specified field of the REC_DATA or REC_TIME data structure is not within the legal range.

[DVD+VR] ERROR **6718** (ref. [DVD+VR] 3.4.1(13/15))
ERR_DVDVR_PCI_GI_REC_TIME_INVALID
REC_TIME_<1 | 2>: Field '<field name>' has an incorrect value <value>. It must be FFh if no information is specified.
When any of the 3 REC_TIME fields "Hour", "Minute" or "Second" specifies the value 'FFh', the REC_TIME data structure is assumed to specify 'No Information', and all REC_TIME fields must code the value 'FFh'.

[DVD+VR] ERROR **6719** (ref. [DVD+VR] 3.4.1(12/14))
ERR_DVDVR_PCI_GI_REC_DATE_TZ_SIGN_ILLEGAL
REC_DATE_<1 | 2>:' TZ Sign' has illegal value <value> because when Time Zone is <value>, the 'TZ Sign' must be 1.

If the value in Time Zone field is '1Fh', the value of TZ Sign must be '1b'.

[DVD+VR] ERROR **6720** (ref. [DVD+VR] 3.4.1(9))
ERR_DVDVR_PCI_GI_CHNG_FLD_VALUE_ILLEGAL
The PCI_GI Extension CHNG_FLD contains an illegal value <value>.
It must be smaller than the number of encoded video fields (<value>) in the current VOBU.

9.4.5.5.2 NSML_AGLI Checks

[DVD+VR] ERROR **6731** (ref. [DVD+VR] 3.3.6)
ERR_DVDVR_PCI_NSML_AGLI_NOT0
PCI NSML_AGLI_C<number>_DSTA : non-zero value <AGL_C location | Destination address of AGL_C> <value>, although there are no Angles allowed in DVD+RW Video.

Since no angle changes are supported by DVD+RW Video, the PCI data may not contain valid addresses for non-seamless angle changes.

9.4.5.5.3 RECI Checks

[DVD+VR] ODDITY **6741** (ref. [DVD+VR] 3.3.7)
ERR_DVDVR_PCI_RECI_ISRC_SP
PCI RECI : ISRC_SP entry <Validity flag | Country Code | Copyright Holder Code | Recording Year | Recording Number> has valid data (<hex value>), while only 1 Sub-picture stream (decoding stream nr 0) is allowed in DVD+RW Video.

One of the RECI ISRC fields for Sub-picture streams 1..31 is not empty, which although not explicitly forbidden, is strange since only 1 Sub-picture stream may be present on DVD+RW Video.

9.4.5.5.4 VOBU_CAT Checks

[DVD+VR] ERROR **6749** (ref. [DVD+VR] 2.2.5)
ERR_DVDVR_PCI_VOBU_CAT
PCI PCI_GI VOBU_CAT specifies non-zero APS trigger bits: The APSTB field has the value <value> (<APS type string>).

The Analogue Protection System may not be used.

9.4.5.6 DVD+RW Video Specific DSI Checks

These checks relate to the DSI data structure.

9.4.5.6.1 DSI_GI Checks

[DVD+VR] ERROR **6751** (ref. [DVD+VR] 3.3.7.1)
ERR_DVDVR_DSI_VOBU_VOB_IDN
DSI_GI : Incorrect VOB ID number VOBU_VOB_IDN (<value>) specified. It must always be set to 1.

In DVD+RW Video all VOBs must have the same VOBU_VOB_IDN value ‘1’.

[DVD+VR] ERROR **6753** (ref. [DVD+VR] 3.3.7.1)
ERR_DVDVR_DSI_VOBU_C_IDN
DSI_GI : Incorrect Cell ID number VOBU_C_IDN <value> specified. It must be the same for all VOBUs belonging to the same Cell (value <value>).

9.4.5.6.2 SML_PBI Checks

[DVD+VR] ERROR **6761** (ref. [DVD+VR] 3.3.7)
ERR_DVDVR_DSI_PREU_ILVU_FLAG
DSI_GI SML_PBI : <PREU_flag | ILVU_flag | Unit_Start_flag | Unit_End_flag> is <value>;
It must be 0 since all VOBs are allocated in Contiguous Blocks, so this VOBU can not be part of a PREU or ILVU.

DVD+RW Video supports no Interleaved Blocks, so these flags must all be set to zero.

[DVD+VR] ERROR **6765** (ref. [DVD+VR] 3.3.7.2.1)
ERR_DVDVR_DSI_VOB_V_S_PTM_1ST
DSI_GI SML_PBI : The VOB_V_S_PTM value <value> (<value> seconds) must be the presentation start time of the first video frame (in display order) in this VOB <value> (<value> seconds), when the VOB's first pack SCR is zero.

When the SCR of the VOB’s first pack is zero, the first packs of the VOBU have not been overwritten. In this case the normal DVD-Video restrictions still hold.

[DVD+VR] ERROR **6766** (ref. [DVD+VR] 3.3.7.2.1)
ERR_DVDVR_DSI_VOB_V_S_PTM
DSI_GI SML_PBI : The VOB_V_S_PTM value <value> is not smaller than the VOBU_S_PTM of the first Cell of this VOB <value>, while the VOB's first pack SCR <value> is not zero.

Since an SCR > 0 indicates that the first packs of the VOB (may) have been overwritten, the (original) VOB start time must be smaller than the start time of its new first VOBU.

This is checked after the DSI data has been parsed completely.
Not the pack SCR but the DSI_GI NV_PCK_SCR is used.

! Remark ! A non-zero SCR does not necessarily imply that the original start of the VOB has been overwritten, because it is only recommended to start a new-recorded VOB with a zero SCR. The DVD+RW Video RTA system may decide to start the VOB of a new recording with a non-zero SCR. But then, it must insert an artificial VOB_V_S_PTM value smaller than the first VOBU_S_PTM.
However, the inverse does hold : when the first packs of a VOB have been overwritten then the SCR > 0.

[DVD+VR] ERROR **6768** (ref. [DVD+VR] 3.3.7.2.1)
ERR_DVDVR_DSI_VOB_V_S_PTM_LIM
SML_PBI : The VOB_V_S_PTM value <value> must be smaller than <hex value>.

VOB_V_S_PTM must always be smaller than ‘20000h’.

[DVD+VR] ERROR **6769** (ref. [DVD+VR] 3.3.7.2.2)
ERR_DVDVR_DSI_VOB_V_E_PTM
DSI_GI SML_PBI : The VOB_V_E_PTM value <value> (<value> seconds) must be equal to the VOB_V_S_PTM <value> + 2^20 video frame presentation periods = <value> (<value> seconds).

VOB_V_E_PTM must be equal to ‘BBB00000h’ in case of NTSC, and ‘E1000000h’ in case of PAL.

9.4.5.6.3 SML_AGLI Checks

[DVD+VR] ERROR **6771** (ref. [DVD+VR] 3.3.7)
ERR_DVDVR_DSI_SML_AGLI_NOT0
DSI SML_AGL_C<number>_DSTA : non-zero value <AGL_C location | Destination address of AGL_C | Size of destination ILVU of AGL_C> <value>, although there are no Angles allowed in DVD+RW Video.

Since no angle changes are supported by DVD+RW Video, the PCI data may not contain valid addresses for non-seamless angle changes.

9.4.5.6.4 VOBU_SRI Checks

[DVD+VR] ERROR **6781** (ref. [DVD+VR] 3.3.7.3)
ERR_DVDVR_DSI_SRI_FBWD_EX_1
VOBU_SRI : FWDI <index> V_FWD_Exist1 flag must be 0 for FWDI 11..240.

[DVD+VR] ERROR **6782** (ref. [DVD+VR] 3.3.7.3)
ERR_DVDVR_DSI_SRI_FBWD_EX_NOTEQ6
VOBU_SRI : FWDI <index> V_FWD_Exist1 flag is <value>;
It must be equal to that of FWDI 6 (<value>) for FWDI 7..10.

[DVD+VR] ERROR **6786** (ref. [DVD+VR] 3.3.7.3)
ERR_DVDVR_DSI_SRI_FBWDA_EXST
VOBU_SRI : FWDI <index> specified VOBU address <value> must be 0x3FFFFFFF for FWDI 11..240.

[DVD+VR] ERROR **6787** (ref. [DVD+VR] 3.3.7.3)
ERR_DVDVR_DSI_SRI_FBWDA_NOTEQ6
VOBU_SRI : FWDI <index> specified VOBU address <value>;
It must be the same as FWDI 6 (<value>) for FWDI 7..10.

9.4.5.6.5 SYNCI Checks

[DVD+VR] ODDITY **6791** (ref. [DVD+VR] 3.3.8)
ERR_DVDVR_DSI_SYNCI_PCK_STRM
SYNCI SP_SYNCA <number>: The <SP_PCK location | SP_PCKA> field (<value>) must be zero, since only 1 Sub-picture stream is allowed in DVD+RW Video.

- This is only reported as an oddity, since it is not explicitly forbidden by the spec.

[DVD+VR] ERROR **6793** (ref. [DVD+VR] (3.3.8))
ERR_DVDVR_DSI_SP_SYNCI_ERR
SYNCI SP_SYNCA <number>: The <SP_PCK location | SP_PCKA> field (<value>) must be <value>, since the data of a Sub-picture stream must be fully contained in one VOBU.

Since the data of a Sub-picture stream must be fully contained in one VOBU, the SP_PCK location flag must be set to '0', indicating that the target SP_PCK is after this NV_PCK, and the SP_PCKA field must contain all '1'.

Disabled DVD Checks

In the DVD specific DSI verification module changes have been made to disable or replace the checks listed below. In case of replacement by a DVD+RW Video variant or to be active for the VMGM_VOBS only, these are implemented or simply copied to a DVD+RW specific verification module.

9.4.5.6.6 Disabled DSI_GI Checks

[DVD] ERROR **4617** (ref. [DVD-3] 4.5.1 (8))
ERR_DVD_DSI_VOBU_C_IDN
DSI_GI : Incorrect Cell ID number VOBU_C_IDN (<value>) specified : must be <equal to or 1 higher | at most 1 higher> than the previous value <value>.

In DVD+RW Video Cells do not have to be numbered consecutively, as long as their number is unique.

9.4.5.6.7 Disabled SML_PBI Checks

[DVD] ERROR **4641** (ref. [DVD-3] 4.5.2 (5))
ERR_DVD_DSI_VOB_V_S_PTM
SML_PBI : The VOB_V_S_PTM value <value> (<value> seconds) must be the presentation start time of the first video frame of the first GOP in this VOB <value> (<value> seconds).

In DVD+RW Video this is only demanded when the start SCR of a VOB is equal to zero. But the latter is no longer mandatory. The RTA system of the DVD+RW Video recorder can choose to start a VOB with a non-zero SCR, e.g. to avoid having to insert the P-STD parameters. But in

this case, VOB_V_S_PTM is even not allowed to be equal to VOBU_S_PTM ! (cf. ERROR **6766**)

[DVD] ERROR **4642** (ref. [DVD-3] 4.5.2 (6))
ERR_DVD_DSI_VOB_V_E_PTM
SML_PBI : The VOB_V_E_PTM value <value> (<value> seconds) must be equal to the presentation termination time of the last video frame of the last GOP in this VOB : <value> (<value> seconds).

[DVD] ERROR **4643** (ref. [DVD-3] 4.5.2 (6))
ERR_DVD_DSI_VOB_V_E_PTM_MULT
SML_PBI : The VOB_V_E_PTM value <value> must be aligned with the video field grid (grid start on <value>, <value> tick period) when the video data does not exist or is terminated.

9.4.5.6.8 Disabled VOBU_SRI Checks

The following checks are disabled only in case of a VTSTT_VOBS for Forward pointers with indexes larger than 6.

[DVD] ERROR **4671**
ERR_DVD_DSI_SRI_FBWD_EX_1
VOBU_SRI : FWDI <index> V_FWD_Exist1 flag <value> specifies incorrectly <non->existing video data in the destination VOBU.

[DVD] ERROR **4673**
ERR_DVD_DSI_SRI_FBWD_EX_2
VOBU_SRI : FWDI <index> V_FWD_Exist2 flag <value> specifies incorrectly <non->existing video data between the VOBU to be presented just after the predecessor at pack <number> and the <VOBU to be presented just before the VOBU addressed by this entry | last VOBU in this Cell> at pack <number>.

[DVD] ERROR **4674**
ERR_DVD_DSI_SRI_FBWD_EX_2_0
VOBU_SRI : FWDI <index> V_FWD_Exist2 flag must be 0 when both the destination VOBU and the predecessor do not exist.

[DVD] ERROR **4677**
ERR_DVD_DSI_SRI_FBWD_EX_1_FLST
VOBU_SRI : FWDI Next V_FWD_Exist1 flag must be 0 for the last VOBU of the Cell.

[DVD] ERROR **4680**
ERR_DVD_DSI_SRI_FBWDA_ILL
VOBU_SRI : FWDI <index> specifies a non-existing VOBU relative address <value>.

[DVD] ERROR **4681**
ERR_DVD_DSI_SRI_FBWDA_ERR
VOBU_SRI : FWDI <index> specifies an incorrect VOBU address <value> (pack <number>); It must be <value> (pack <number>).

[DVD] ERROR **4682**
ERR_DVD_DSI_SRI_FBWDA_S_PTM
VOBU_SRI : FWDI <index> addresses a VOBU presented from <value> to <value> which is not at an offset of <value> times 0.5 sec from the presentation start time (<value>) of the VOBU containing this DSI, which is <value>.

[DVD] ERROR **4683**
ERR_DVD_DSI_SRI_FBWDA_ERRPOS
VOBU_SRI : FWDI <index> specifies a VOBU at address <value> which is after the Cell start at <value>.

[DVD] ERROR **4684**
ERR_DVD_DSI_SRI_FBWDA_EXST
VOBU_SRI : FWDI <index> specified VOBU address <value> must be 0x3FFFFFFF for a non-existing VOBU.

[DVD] ERROR **4685**
ERR_DVD_DSI_SRI_FBWDA_N_EXST
VOBU_SRI : FWDI <index> specified a non-existing destination VOBU at time <value>, while a VOBU being presented from <value> to <value> exists at pack <number>.

[DVD] ERROR **4687**
ERR_DVD_DSI_SRI_FBWDA_EXST_FLST
VOBU_SRI : FWDI Next specified VOBU address <value> must be 0x3FFFFFFF for the last VOBU of a Cell.

[DVD] ERROR **4689**
ERR_DVD_DSI_SRI_FBWD_NOPRED
VOBU_SRI : FWDI <index> predecessor not found !

9.4.5.6.9 Disabled SYNCI Checks

[DVD] ERROR **4693**
ERR_DVD_DSI_SYNCI_PCK_EXST
SYNCI SP_SYNCA <number> : A zero SP_PCKA field indicates this Sub-picture stream to be non-present, while there are <value> Sub-picture streams.

[DVD] ERROR **4698**
ERR_DVD_DSI_SYNCI_SP_PCKA_E
SYNCI SP_SYNCA <number> : The SP_PCKA target address is <value>; It must have all bits set when the target SP_PCK does not exist (SP_PCK location flag zero).

9.4.6 VRMI Data Checks

The module should perform all checks on the recording data present on a DVD+RW Video disc. The required checks, as derived from the DVD+RW Video specification are listed below. Unless explicitly stated otherwise, these are all reported as errors.

9.4.6.1 Generic Checks

[DVD-VR] ERROR **6801** (ref. [DVD_VR] 4, 4.2, 4.3, 4.4)
ERR_DVDVR_VRMI_RESERVED_ILLEGAL
VRMI: The reserved field of '<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be <value>.

All reserved fields must be 'zero'.

[DVD-VR] ERROR **6802** (ref. [DVD-VR] 4.2(76), 4.4.3(7), 4.4.3(12))
ERR_DVDVR_VRMI_BCD_ILLEGAL
VRMI: The '<field>' in '<vrmi structure name>' of '<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> - an invalid BCD value.

[DVD-VR] ERROR **6803** (ref. [DVD-VR] 4.2 (38), 4.4.3(16), DVD-3 4.3.2 (2))
ERR_DVDVR_VRMI_RANGE_ILLEGAL
VRMI: The '<field>' in '< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be between [<value> … <value>].

[DVD-VR] ERROR **6804** (ref. [DVD-VR] 4.2(38), 4.2(40), 4.2(80))
ERR_DVDVR_VRMI_VALUE_ILLEGAL
VRMI: The '<field>' in '< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be <value>.

9.4.6.2 Date Checks

[DVD-VR] ERROR **6805** (ref. [DVD-VR] 4.2(71), 4.4.3(7))
ERR_DVDVR_VRMI_DATE_RANGE_ILLEGAL
VRMI: The '<field>' of DATE in '< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be between [<value> … <value>].

Year field must have values in the range of '2000 '- '9999'.
Month field must have values in the range '01' - '12'
Day field must have values in the range '01' - '31'.
Time Zone field must have values in the range '00' - '14'.

[DVD-VR] ERROR **6806** (ref. [DVD-VR] 4.2(71), 4.4.3(7))
ERR_DVDVR_VRMI_DATE_RESERVED
VRMI: The '<field>' of DATE in '< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' has a reserved value <value>.

Values other than 'FFFFh' or the range '2000'-'9999' for the Year field are reserved.
Values other than '1Fh' or the range '01-'12 for the Month field are reserved. Values other than '3Fh' or the range '01' - '31' for the Day field are reserved.
Values other than '1Fh' or the range '00' - '14' for the Time Zone field are reserved.

[DVD-VR] ERROR **6807** (ref. [DVD-VR] 4.2(71), 4.4.3(7))
ERR_DVDVR_VRMI_DATE_INVALID_VALUE
VRMI: The '<field>' of DATE in '< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be <value> if Year has <value>.

If the Year field has the value ‘FFFFh’, all other bits of DATE must be ’1b’ with the exception of reserved fields.

[DVD-VR] ERROR **6808** (ref. [DVD-VR] 4.2(71), 4.4.3(7))
ERR_DVDVR_VRMI_DATE_INVALID_DSTMTZ
VRMI: The ‘<field>’ of DATE in ‘< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>’ is <value> but must be <value> because Time Zone is <value>.

The fields DS, TM & TZ Sign must be ‘1b’ if Time Zone has value ‘1Fh’.

9.4.6.3 Time Checks

[DVD-VR] ERROR **6810** (ref. [DVD-VR] 4.2(76), 4.4.3(12))
ERR_DVDVR_VRMI_TIME_RANGE_ILLEGAL
VRMI: The ‘<field>’ of TIME in ‘< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>’ is <value> but must be between [<value> … <value>].

Hour fields must have a value in the range ‘00’-‘23’.
Minute fields must have a value in the range ‘00’-‘59’.
Seconds fields must have a value in the following range ‘00’-‘59’.

[DVD-VR] ERROR **6811** (ref. [DVD-VR] 4.2(76), 4.4.3(12))
ERR_DVDVR_VRMI_TIME_RESERVED
VRMI: The ‘<field>’ of TIME in ‘< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>’ has reserved <value>.

Valid values for the Hour field are the range ‘00’-‘23’ or ‘FFh’. All other values are reserved.
Valid values for the Minute field are the range ‘00’-‘59’ or ‘FFh’. All other values are reserved.
Valid values for the Seconds field are the range ‘00’-‘59’ or ‘FFh’. All other values are reserved.

[DVD-VR] ERROR **6812** (ref. [DVD-VR] 4.2(76), 4.4.3(12))
ERR_DVDVR_VRMI_TIME_INVALID_VALUE
VRMI: The ‘<field>’ of TIME in ‘< VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>’ is <value> but must be <value> if the Hour field has <value>.

If Hour field contains the value ‘FFh’, the Minute and Second fields must also contain the value ‘FFh’.

9.4.6.4 Key Frame Checks

[DVD-VR] ERROR **6814** (ref. [DVD-VR] 4.2(116), 4.3(32-8), 4.4.3(52))
ERR_DVDVR_VRMI_KF_VIDEO_FRAME_NUMBER_INVALID
VRMI: The ‘<field>’ of Key Frame Pointer structure in ‘<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>’ is <value> but must be greater than or equal to 1 because counting of video frames starts from 1 at the beginning of video presentation of a VOBU.

KF_Video frame_number is an unsigned integer indicating which of the video frames within the presentation period of the VOBU referred by KF_VOBU_A is the key frame for the disc.

[DVD-VR] ERROR **6815** (ref. [DVD-VR] 4.2(116) /4.3(32-8)/4.4.3(52))
ERR_DVDVR_VRMI_KF_VOBU_A_ILLEGAL
VRMI: The KF_VOBU_A of Key Frame Pointer structure in ‘<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>’ is <value> but must be 0 if no key frame is selected.

9.4.6.5 Name Format Checks

[DVD-VR] ERROR **6817** (ref. [DVD-VR] 4.2(124), 4.4.3(60))
ERR_DVDVR_VRMI_NM_FMT_CHAR_SET_RESERVED
VRMI: The '<field>' of Name format structure in '<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' has the reserved value <value>. Allowed values for 'Char_Set_1' is 0x11h and for 'Char_Set_2' 0x12h, 0x13h and 0x21h.

The Char_Set_1 must have the value '11h'. All other values are reserved.
The Char_Set_2 must have values '12h', '13h' or '21h'. All other values are reserved.

[DVD-VR] ERROR **6818** (ref. [DVD-VR] 4.2(124), 4.4.3(60))
ERR_DVDVR_VRMI_NM_FMT_LENGTH_RANGE_ILLEGAL
VRMI: The '<field>' of Name format structure in '<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be in the range [<value> … <value>].

Length 1 field of NM_FMT must be in the range [0…64].
Length 2 field of NM_FMT must be in range [0…64].

[DVD-VR] ERROR **6819** (ref. [DVD-VR] 4.2(124), 4.4.3(60))
ERR_DVDVR_VRMI_NM_FMT_LENGTH_2_NOT_EVEN
VRMI: The Length_2 of Name format structure in '<VRMI_GI|VRMI_CHPI|VRMI_RECI|VRMI_RES>' is <value> but must be even because Char_Set_2 specifies a double byte character set code.

If Char_Set_2 specifies a double byte character set code, Length_2 shall be even.

9.4.7 VRMI_GI checks

[DVD-VR] ERROR **6821** (ref. [DVD-VR] 4.2(0))
ERR_DVDVR_VRMI_GI_VRM_ID_ILLEGAL
VRMI_GI: The VRM_ID string specified is '<value>' but must be '<value>'

The VRM_ID field must contain the string "DVDVRMANAGER".

[DVD-VR] ERROR **6822** (ref. [DVD-VR] 4.2(28))
ERR_DVDVR_VRMI_GI_VRMI_EA_ILLEGAL
VRMI_GI: The VRMI_EA address specified is <hex value> but must be <hex value>.

For the current version (0.9) of the DVD_VR specification, VRMI_EA must have the value '0010h'.

[DVD-VR] ERROR **6823** (ref. [DVD-VR] 4.2(32))
ERR_DVDVR_VRMI_GI_VERN_ILLEGAL
VRMI_GI: The VERN number specified is <hex value> but must be <hex value>

For the current version (3.0) of the DVD_VR specification, VERN must have the value '0030h' indicating Version 3.0

[DVD-VR] ERROR **6824** (ref. [DVD-VR] 4.2(64))
ERR_DVDVR_VRMI_GI_DSC_ST_TVSYSTEM_RESERVED
VRMI_GI: The DSC_ST TV System specified has reserved <value> but must be 0 for NTSC or 1 for PAL.

The TV System must have the value '00b' for NTSC or '01b' for PAL.

[DVD-VR] ERROR **6825** (ref. [DVD-VR] 4.2(128))
ERR_DVDVR_VRMI_GI_DSC_NM_INVALID
VRMI_GI: The DSC_NM[<index>] in VRMI_GI unused byte is <value> but must be 0 because all unused bytes must be 0. The number of used bytes is <value>.

Unused bytes contain value ‘00h’.

[DVD-VR] ERROR **6826** (ref. [DVD-VR] 4.2(192))
ERR_DVDVR_VRMI_GI_ALT_DSC_NM_INVALID
VRMI_GI: The ALT_DSC_NM[<index>] in VRMI_GI unused byte is <value> but must be 0 because all unused bytes must be 0. The number of used bytes is <value>.

Unused bytes contain value ‘00h’.

[DVD-VR] ERROR **6827** (ref. [DVD-VR] 4.2(256))
ERR_DVDVR_VRMI_GI_MAN_ID_ILLEGAL_CHARACTER
VRMI_GI: The MAN_ID[<index>] character is <value> but must be between [0x20h …0x7Eh] because all coded characters are ISO/IEC 8850-1.

[DVD-VR] ERROR **6828** (ref. [DVD-VR] 4.2(256))
ERR_DVDVR_VRMI_GI_MAN_ID_INVALID
VRMI_GI: The MAN_ID[<index>] unused byte is <value> but must be 0 because all unused bytes must be 0.

Unused bytes contain value ‘00h’.

[DVD-VR] ERROR **6829** (ref. [DVD-VR] 4.2(288))
ERR_DVDVR_VRMI_GI_MDL_ID_ILLEGAL_CHARACTER
VRMI_GI: The MDL_ID[<index>] character is <value> but must be in the range [0x20h…0x7Eh] because all coded characters are ISO/IEC 8850-1.

[DVD-VR] ERROR **6830** (ref. [DVD-VR] 4.2(288))
ERR_DVDVR_VRMI_GI_MDL_ID_INVALID
VRMI_GI: The MDL_ID[<index>] unused byte is <value> but must be 0 because all unused bytes must be 0.

Unused bytes contain value ‘00h’.

[DVD-VR] ERROR **6831**(ref. [DVD-VR] 4.2(304))
ERR_DVDVR_VRMI_GI_FRMW_ILLEGAL_CHARACTER
VRMI_GI: The FRMW_ID[<index>] character is <value> but must be in the range [0x20h…0x7Eh] because all coded characters are ISO/IEC 8850-1.

[DVD-VR] ERROR **6832** (ref. [DVD-VR] 4.2(304))
ERR_DVDVR_VRMI_GI_FRMW_ID_INVALID
VRMI_GI: The FRMW_ID[<index>] unused byte is <value> but must be 0 because all unused bytes must be 0.

Unused bytes contain value ‘00h’.

Note: MENU_LO_ID Is defined but has no interesting verifiable information. Hence the checks for MENU_LO_ID field have not been implemented.

[DVD-VR] ERROR **6833** (ref. [DVD-VR] 4.2(1024))
ERR_DVDVR_VRMI_GI_FTT_NS_INVALID
VRMI_GI: The FTT_Ns has <value> but must be in the range [1…49].

This field indicates the number of full titles in DVD-Video zone.

[DVD-VR] ERROR **6834** (ref. [DVD-VR] 4.2(1025))
ERR_DVDVR_VRMI_GI_DSC_PB_NS_INVALID
VRMI_GI: The DSC_PB_Ns is <value> but must be between [0…99].

[DVD-VR] ERROR **6835** (ref. [DVD-VR] 4.2(34))
ERR_DVDVR_VRMI_GI_VR_APP_RESERVED
VRMI_GI: The VR_APP specified has reserved value <value> but must be 0x00h or 0x01h.

[DVD-VR] ERROR **6836** (ref. [DVD-VR] 4.2(1026))
ERR_DVDVR_VRMI_GI_DSC_PB_SEQT_INVALID
VRMI_GI: The DSC_PB_SEQT [<index>] is <value> but must be 0 if DSC_PB_Ns is 0.

When DSC_PB_Ns has value '00h', all DSC_PB_SEQT entries are 'zero'.

[DVD-VR] ERROR **6837**(ref. [DVD-VR] 4.2(1026))
ERR_DVDVR_VRMI_GI_DSC_PB_SEQT_ILLEGAL
VRMI_GI: The unused entry of DSC_PB_SEQT[<index>] is <value> but must be 0 because all unused bytes must be 0. The number of used entries is <value>.

Unused entries at the end contain the value '00h'.

[DVD-VR] ERROR **6838**(ref. [DVD-VR] 4.2(0))
ERR_DVDVR_VRMI_GI_VRM_ID_ILLEGAL_CHARACTER
VRMI_GI: The VRM_ID[<index>] is <value> but must be in the range[<value>…. <value>] because all characters are of the ISO-646 character set.

The characters have to be in the range 20h…7Eh.

[DVD-VR] ERROR **6839**(ref. [DVD-VR] 4.2(1026))
ERR_DVDVR_VRMI_GI_DSC_PB_SEQT_ENTRY_ILLEGAL
VRMI_GI: The DSC_PB_SEQT[<index>] is <value> which is a deleted recording because the VRMI_RECI[<index>]'s REC_ST=%d and hence may not be referenced.

Deleted recordings may not be referenced.

DVD-VR] ERROR **6840** (ref. [DVD+VR] 4.2)
ERR_DVDVR_VRMI_GI_CELL_INFO_FLG_VERN
VRMI_GI: VERN indicates that this is a 3.x disc. If this is a 3.x disc the disc shall have CELL_INFO_FLG = 01h.

9.4.8 VRMI CHPI Checks

[DVD-VR] ERROR **6841** (ref. [DVD_VR] 1.5.2)
ERR_DVDVR_VRMI_CHPI_VRCHP_CELLID_ILL
VRMI_CHPI: The CELL_ID=<value> is illegal because this ID is reserved for buffer cells.

CELL_ID number 255 is reserved for exclusive use of buffer cells. The buffer cell may not be used by any program chain.

[DVD-VR] ERROR **6842** (ref. [DVD_VR]1.5.32/DVD-3 2.4.100)
ERR_DVDVR_VRMI_CHPI_VOBU_SIZE_ILLEGAL_VALUE
VRMI_CHPI: The VOBU_SIZE <value> of VRCHP_IT[<index>] is illegal. It must be at least two because a VOBU must have a NV_PCK and at least one encoded video frame.

[DVD-VR] ERROR **6843** (ref. [DVD-VR] 4.3(0))
ERR_DVDVR_VRMI_CHPI_VRCHP_Ns_ILLEGAL_VALUE
VRMI_CHPI: The VRCHP_Ns is <value> but must be in the range [<value> … <value>].

Valid VR chapter markers are in the range [0..254].

[DVD-VR] ERROR **6844** (ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_VRCHP_MKI_CELLVOBUPIC_ILLEGAL
VRMI_CHPI: The VRCHP_MKI's Cell_start <value>, VOBU_start <value> and Video frame Number <value> is illegal when Rec_start is 1.

If Rec_start is set to '1', Cell_start and VOBU_start must be '1' and Video frame Number must contain the value '1'.

[DVD-VR] ERROR **6845** (ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_VRCHP_MKI_VOBUPIC_ILLEGAL
VRMI_CHPI: The VRCHP_MKI's VOBU_start <value> and Video frame Number <value> is illegal when Cell_start is <value>.

If Cell_start is set to 1 VOBU_start must be 1 and Video frame Number must contain the value 1.

[DVD-VR] ERROR **6846** (ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_VRCHP_MKI_PIC_ILLEGAL
VRMI_CHPI: The VRCHP_MKI's Video frame Number <value> is illegal when VOBU_start is <value>.

If VOBU_start is 1, Video frame Number must contain the value 1.

[DVD-VR] ERROR **6847**(ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_VRCHP_MKI_VOBUA_DUPLICATE
VRMI_CHPI: The VRCHP_IT[<index>] VRCHP_MKI CHP_VOBU_A is identical to that of VRCHP_IT[<index>] which is not allowed since no two CHP_VOBU_A fields in VRMI_CHPI shall be identical.

No two CHP_VOBU_A fields in VRMI_CHPI shall contain the same value.

[DVD-VR] ERROR **6848**(ref. [DVD-VR] 4.3(32-8))
ERR_DVDVR_VRMI_CHPI_VRCHP_KFI_VOBUA_ILLEGAL
VRMI_CHPI: The VRCHP_KFI's of VRCHP_IT[<index>] KF_VOBU_A is <value> but must be greater than or equal to VRCHP_IT[<index>] CHP_VOBU_A <value> and less than the next table entry VRCHP_IT[<index>] CHP_VOBU_A <value>.

[DVD-VR] ERROR **6849**(ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_CHP_VOBU_A_NOT_INCREMENTED
VRMI_CHPI: The VRCHP_IT[<index>] VRCHP_MKI CHP_VOBU_A is <value>, but it must be greater than the CHP_VOBU_A <value> of VRCHP_IT[<index>] because VRCHPI blocks must be encoded in order of increasing CHP_VOBU_A.

VRCHPI blocks start from first byte of the table in order of increasing CHP_VOBU_A.

[DVD-VR] ERROR **6850**(ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_VRCHP_VOBU_A_NO_REC_START
VRMI_CHPI: The VRCHP_IT[<index>] VRCHP_MKI CHP_VOBU_A is <value>. However this does not mark the start of the first VOBU of any non-deleted VRMI_RECI, which requires attention since the Rec start flag is set. (VRMI_RECI #<value> starts at <value> and ends at <value>). The start of the VRMI_RECI is the previous VRMI_RECI.REC_E_A+1.

[DVD-VR] ERROR **6851**(ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_CHPI_VRCHP_MKI_VFNUM_INVALID
VRMI_CHPI: The '< value>' in '<value>[<index>]' is <value> but must be greater than or equal to 1 because counting of video frames starts from 1.

This check is applied to the Video_frame_number field of VRCHP_MKI

[DVD-VR] ERROR **6852**(ref. [DVD-VR] 4.3(32-0/8))
ERR_DVDVR_VRMI_CHPI_VRCHP_IT_ILLEGAL
VRMI_CHPI: The '<VRCHP_MKI | VRCHP_KFI>' of VRCHP_IT[<index>] is non-zero. It must be 'zero' because it is an unused entry and all unused entries must have the value 'zero'.

[DVD-VR] ERROR **6853**(ref. [DVD-VR] 4.3(0))
ERR_DVDVR_VRMI_CHPI_VRCHP_NS_INVALID
VRMI_CHPI: VRCHP_Ns is <value> of VRMI_CHPI but must be 0 if no video has been recorded on disc or if all video has been deleted

9.4.9 VRMI RECI Checks

[DVD-VR] ODDITY **6860**(ref. [DVD-VR] 4.4.2(0))
ERR_DVDVR_VRMI_RECI_IS_NOT_EMPTY
VRMI_RECI: The <value> in REC_ST of VRMI_RECI[<index>] indicates an Empty VRMI_RECI. But VRMI_RECI has non-zero values in the reserved fields.

[DVD-VR] ERROR **6861**(ref. [DVD-VR] 4.4.3(1))
ERR_DVDVR_VRMI_RECI_REC_MODE_PROTECTION_RESERVED
VRMI_RECI: The <value> of Protection in REC_MODE of VRMI_RECI[<index>] is reserved. The allowed values are 0 and 7.

[DVD-VR] ERROR **6862**(ref. [DVD-VR] 4.4.3(1))
ERR_DVDVR_VRMI_RECI_REC_MODE_PGMATCH_RESERVED
VRMI_RECI: The <value> of PG Match in REC_MODE of VRMI_RECI[<index>] is reserved. The allowed values are 0 and 1.

[DVD-VR] ERROR **6863**(ref. [DVD-VR] 4.4.3(3))
ERR_DVDVR_VRMI_RECI_REC_BRT_RESERVED
VRMI_RECI: The <value> of Bitrate mode in REC_BRT in REC_MODE of VRMI_RECI[<index>] is reserved. The allowed values are 0, 1,2,3 and 4.

[DVD-VR] ERROR **6864**(ref. [DVD-VR] 4.4.3(3))
ERR_DVDVR_VRMI_RECI_REC_INDEX_ILLEGAL
VRMI_RECI: The VRMI_RECI[<index>] REC_BRT Bitrate index has illegal value <value>. It must be <value> for Bitrate mode <value>.

[DVD-VR] ERROR **6865**(ref. [DVD-VR] 4.4.3(4))
ERR_DVDVR_VRMI_RECI_REC_SRC_RESERVED
VRMI_RECI: The REC_SRC of VRMI_RECI[<index>] is <value> which is reserved. The allowed values are 00h, 01h, 02h, 03h, 11h, 12h, 13h and FFh.

Note: All values are allowed for REC_CNT.
Note: REC_DATE has the same possible error messages as LAST_DATE in VRMI_GI
Note: REC_TIME has the same possible error messages as LAST_TIME in VRMI_GI

[DVD-VR] ERROR **6866**(ref. [DVD-VR] 4.4.3(64))
ERR_DVDVR_VRMI_RECI_REC_NM_INVALID
VRMI_RECI: The REC_NM of VRMI_RECI[<index>] unused byte is <value> but must be 00h because all unused bytes must be 00h. The number of used bytes is <value>.

Unused bytes contain value '00h'.

[DVD-VR] ERROR **6867** (ref. [DVD-VR] 4.4.3(128))
ERR_DVDVR_VRMI_RECI_ALT_REC_NM_INVALID
VRMI_RECI: The ALT_REC_NM of VRMI_RECI[<index>] unused bytes is <value> but must be 00h because all unused bytes must be 00h. The number of used bytes is <value>.

Unused bytes contain value '00h'.

[DVD-VR] ERROR **6868** (ref. [DVD-VR] 3.2.4)
ERR_DVDVR_VRMI_RECI_CONSECUTIVE_DELETED_RECORDING
VRMI_RECI: Found two consecutive deleted recordings at VRMI_RECI[<index>] and VRMI_RECI[<index>]. When two consecutive recordings are deleted, they shall be combined to one deleted recording.

[DVD-VR] ERROR **6869** (ref. [DVD-VR] 3.2.4)
ERR_DVDVR_VRMI_RECI_LAST_DELETED_RECORDING
VRMI_RECI: Last recording in VTSTT_VOBS is a deleted recording. The last recording shall never be a deleted recording in VTSTT_VOBS.

[DVD-VR] ERROR **6870** (ref. [DVD-VR] 4.4.1)
ERR_DVDVR_VRMI_RECI_EMPTY_RECORDING_NOT_AT_END
VRMI_RECI: Found an empty recording at VRMI_RECI[<index>] although VRMI_RECI[<index>] and VRMI_RECI[<index>] are not empty. Empty VRMI_RECI blocks shall be added at the end until there are 49 VRMI_RECI blocks in total.

Empty VRMI_RECI blocks shall only be added at the end until there are 49 VRMI_RECI blocks in total.

[DVD-VR] ERROR **6871** (ref. DVD_VR 4.4.4(28))
ERR_DVDVR_VRMI_RECI_EA_NOT_ASCENDING
VRMI_RECI: The REC_E_A of VRMI_RECI[<index>] is <value> but must greater than VRMI_RECI<index>]'s EA <value> because REC_E_A values for existing recordings shall be in ascending order.

[DVD-VR] ERROR **6872** (ref. [DVD-VR] 4.4.3(192))
ERR_DVDVR_VRMI_RECI_TOTAL_VRPL_NS_LIM
VRMI_RECI: The sum of all VRMI_RECI VRPL_Ns is <value>;

But it must be at most <value>.

[DVD-VR] ERROR **6873** (ref. [DVD-VR] 4.4.3(192))
ERR_DVDVR_VRMI_RECI_PL_MODE_NOT_0
VRMI_RECI: VRMI_RECI[<index>] PL mode is <value>;
But since VRPL_Ns is <value>, it must be 0.

[DVD-VR] ERROR **6873** (ref. [DVD-VR] 4.4.3(194))
ERR_DVDVR_VRMI_RECI_PL_MODE_NOT_0
VRMI_RECI: The VRPL[<index>] is <value> of VRMI_RECI[<index>] but must be in the range [0 …(REC_VRCHP_Ns-1)].

[DVD-VR] ERROR **6874** (ref. [DVD-VR] 4.4.3(194))
ERR_DVDVR_VRMI_RECI_VRPL_ILLEGAL_VALUE
VRMI_RECI: The VRPL[<index>] is <value> of VRMI_RECI[<index>] but must be in the range [0 …(REC_VRCHP_Ns-1)].

[DVD-VR] ERROR **6875** (ref. [DVD-VR] 4.4.3(194))
ERR_DVDVR_VRMI_RECI_NUM_VRPL_ENCODED_ILLEGAL
VRMI_RECI: The number of VRPL of VRMI_RECI[<index>] encoded is <value> which is higher than the value <value> specified by VRPL_Ns.

[DVD-VR] ERROR **6876** (ref. [DVD-VR] 4.4.3(194))
ERR_DVDVR_VRMI_RECI_VRPL_NOT_INCREMENTED
VRMI_RECI: The VRMI_RECI[<index>] chapter number in VRPL[<index>] is <value> but it must be greater than <value> because, if PL_mode is 1, the chapter references must be in ascending order.

If PL_mode is ‘0b’, chapter references are in ascending order.
If PL_mode is ‘1b’, chapter references may be in arbitrary order and chapters may be referenced more than once.

[DVD-VR] ERROR **6877** (ref. [DVD-VR] 4.4.3(194))
ERR_DVDVR_VRMI_RECI_VRPL_DUPLICATE
VRMI_RECI: The VRMI_RECI[<index>] chapter number in VRPL[<index>] is <value>, but it has already been referenced in entry <value> which is not allowed since PL mode is set to <value>.

[DVD-VR] ERROR **6878** (ref. [DVD-VR] 4.4.3(32))
ERR_DVDVR_VRMI_RECI_VRCHP_NS_RANGE_ILLEGAL
VRMI_RECI: The VRMI_RECI[<index>] VRPL_Ns is <value> but must be between [<value>...<value>].

[DVD-VR] ERROR **6879** (ref. [DVD-VR] 4.4.3(192))
ERR_DVDVR_VRMI_RECI_VRPL_NS_RANGE_ILLEGAL
VRMI_RECI: VRMI_RECI[<index>] VRPL_Ns is <value> but must be in the range [<value>….<value>].

[DVD-VR] ERROR **6880** (ref. [DVD-VR] 4.4.3(194))
ERR_DVDVR_VRMI_RECI_VRPL_UNUSED
VRMI_RECI: The chapter number VRPL[<index>] of VRMI_RECI[<index>] is unused and must be FFh but is <value>.

[DVD-VR] ERROR **6881** (ref. [DVD-VR] 4.4.3(3))
ERR_DVDVR_VRMI_RECI_REC_INDEX_RANGE_ILLEGAL
VRMI_RECI: The VRMI_RECI[<index>] REC_BRT Bitrate index has the illegal value <value>. It must be in the range [<value>..<value>] for Bitrate mode <value>.

[DVD-VR] ERROR **6882** (ref. [DVD-VR] 4.4.2(0), 4.4.3(0), 4.4.4(0))
ERR_DVDVR_VRMI_RECI_REC_ST_ILLEGAL
VRMI_RECI: The REC_ST <value> of VRMI_RECI[<index>] is illegal. Allowed values are 00h, 01h and 02h.

[DVD-VR] ERROR **6883** (ref. [DVD-VR] 4.4.3(52))
ERR_DVDVR_VRMI_RECI_KF_VOBU_A_ILLEGAL
VRMI_RECI: The VRMI_RECI[<index>] has a KF_VOBU_A <value> which is illegal. It must be greater than REC_E_A <value> of a previous existing recording and lesser than REC_E_A <value> of current recording.

[DVD-VR] ERROR **6884** (ref. [DVD-VR] 4.4.3(16)/DVD-3 4.3.2 (2))
ERR_DVDVR_VRMI_RECI_TC_FLAG_RESERVED
VRMI_RECI: The VRMI_RECI[<index>] REC_PB_TM tc_flag has reserved value <value>.The allowed values are 01b and 11b.

[DVD-VR] ERROR **6885** (ref. [DVD-VR] 4.4.4(28))
ERR_DVDVR_VRMI_RECI_DELREC_EA_ILLEGAL
VRMI_RECI: The End Address of VRMI_RECI[<index>] is <value> (<hex value>) is illegal. The REC_E_A for a deleted recording in between existing recordings can never be 'zero'.

[DVD-VR] ERROR **6886** (ref. DVD+VR 4.4.3(6))
ERR_DVDVR_VRMI_RECI_CP_STAT_ILLEGAL
VRMI_RECI: The field CP_stat(<value>) of REC_VOB_IFO of VRMI_RECI[<value>] is illegal, and must be (00b) when CP_METHOD in VRMI_GI is (00h).

[DVD-VR] ERROR **6888** (ref. [DVD+VR] 4.2)
ERR_DVDVR_VRMI_GI_CELL_INFO_FLG_CP_METHOD
VRMI_GI: VERN indicates that this is a 3.x disc. If this is a 3.x disc the disc shall have CP_METHOD = 02h.

[DVD-VR] ERROR **6889** (ref. [DVD+VR] 4.2)
ERR_DVDVR_VRMI_GI_CELL_INFO_TABLE
VRMI_GI: The CELL_INFO[<value>] table contains values other then 00h or 01h.

[DVD-VR] ERROR **6890** (ref. [DVD+VR] 4.2)
ERR_DVDVR_VRMI_GI_CP_METHOD
VRMI_GI: CP_METHOD contains an illegal value <value>, CP_Method should contain 00h, 01h or 02h.

[DVD-VR] ERROR **6891** (ref. [DVD_VR] 4.3(32-0))
ERR_DVDVR_VRMI_VRCHP_MKI_REC_START_INVALID
VRMI_RECI: Found <value> chapters with Rec_Start bit set for VRMI_RECI[<index>].There must be one and only one VRCHP_MKI with Rec_Start bit set to 1 for every existing recording.

[DVD-VR] ERROR **6892** (ref. [DVD-VR] 4.3(0),4.4.3(32))
ERR_DVDVR_CHAPTER_MARKER_NUMBER_MISMATCH
VRMI_RECI: The number of VR Chapter markers on disc in VRMI_CHPI <value> does not match the sum of recorded VR chapter markers <value> for existing recordings

[DVD-VR] ERROR **6893** (ref. [DVD-VR] 4.3(32-0))
ERR_DVDVR_VRMI_VRCHP_MKI_REC_START_ILLEGAL
VRMI_RECI: The Rec_Start bit of VRCHP_IT<index>] is <value> but must be <value> as it is the first chapter marker of VRMI_RECI[<index>].

[DVD-VR] RECOMMENDATION VIOLATION **6894** (ref. [DVD-VR] D.2)
ERR_DVDVR_VRMI_CHPI_VRCHP_Ns_RECOMMEND_ILL
VRMI_RECI: The VRCHP_Ns is <value> but should be smaller than or equal to <value>.

This is a recommendation violation message.

[DVD-VR] ERROR **6895** (ref. [DVD-VR] 4.2 (28))
ERR_DVDVR_EA_ERROR
VRMI: The specified '<value>'s' end address (<hex value>) is not consistent with the parsed length (<hex value>).

[DVD-VR] ERROR **6896** (ref. DVD+VR 4.4.3(6))
ERR_DVDVR_VRMI_RECI_CP_STAT_CELL_INFO_ILLEGAL
VRMI_RECI: The field CP_stat(<hex value>) of REC_VOB_IFO of VRMI_RECI[<hex value>] is illegal, and must match the CELL_INFO value [<hex value>] for cell <hex value>.

9.4.10 Data Zone Layout and File System Checks

Below all checks are listed on a DVD+RW disc's Data Zone layout and File System constraints as described in 2.3 and 2.4 of the DVD+RW Video specification.
Unless explicitly stated otherwise, these are all reported as errors.

Furthermore, also DVD-Video checks that should be disabled for a DVD+RW disc, since the constraints they verify have been removed or relaxed, are listed here.

9.4.10.1 Data Zone Layout and Data Files Allocation

These checks relate to the overall DVD+RW disc data layout and the way, i.e. size, location, order, etc., the data files have to be recorded on the disc.

[DVD+VR] ERROR **6901** (ref. [DVD+VR] 2.3.3)
ERR_DVDVR_VRM_SCRATCH_MIS
VRM Scratch area (file <name>) is missing.

The mandatory VRM Scratch file is not recorded in the file system.

[DVD+VR] ERROR **6902** (ref. [DVD+VR] 2.3.3)
ERR_DVDVR_VRM_SCRATCH_SIZE
VRM Scratch area (<file name>) size is <value> bytes; It must be exactly <value> bytes (1 MB).

The VRM Scratch file does not have the correct (fixed) file size of 1 MByte.

[DVD+VR] ERROR **6905** (ref. [DVD+VR] 2.3.4)
ERR_DVDVR_VRMI_MISSING
VRMI data file <file name> is missing on the disc.

The mandatory VRMI data file is not recorded in the file system.

[DVD+VR] ERROR **6906** (ref. [DVD+VR] 4.1)
ERR_DVDVR_VRMI_SIZE
VRMI (file <name>) size is <value> bytes; It must be exactly <value> bytes (32 kB).

The VRMI data file does not have the expected (fixed) size of exactly 32 kByte.

[DVD+VR] ERROR **6907** (ref. [DVD+VR] 2.3.4, 2.3.7)
ERR_DVDVR_VRMI_MISMATCH
Byte <number> of the VRMI backup file (VIDEO_RM.BUP) with value <value> does not match byte <number> of the VRMI Information file (VIDEO_RM.IFO) having the value <value>.

The VRMI backup data file is not an exact copy of the original version.

[DVD+VR] ERROR **6909** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VIDEO_RM_DIR_ILL
The VIDEO_RM directory contains file <name>; However VRM User Data files with names starting with 'VIDEO_R' are reserved for future use.

A file other than the ones allowed (VRMI and VRM Scratch files) and with a reserved name is stored in the VIDEO_RM directory.

[DVD+VR] ERROR **6910** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_FILE_SIZE_0
Mandatory file <name> has zero size!

One of the mandatory files (VIDEO_TS.IFO, VIDEO_TS.VOB or VTS_0x_0.IFO) is specified by the file system (as required), but with a zero file size which is not allowed.

[DVD+VR] ERROR **6911** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VTS_FILE_MISS
Mandatory file <name> is missing.

One of the mandatory files is missing:
- This can be one of VIDEO_TS.IFO, VIDEO_TS.VOB, VIDEO_TS.BUP, VTS_01_0.IFO or VTS_01_0.BUP
- Or a “VTS_0x_0.BUP” file for which the matching IFO file “VTS_0x_0.IFO” is present.

[DVD+VR] ERROR **6912** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VTSI_NO_VOBS
VTS <number> VTSI <file name> is present on disc, while the matching VTSTT_VOBS <file name> is not.

When the VTSI “VTS_0x_0.IFO” of one of the three possible VTSs is stored on disc, the matching Title VOBS VTSTT_VOBS “VTS_0x_y.VOB” must also be recorded.

[DVD+VR] ERROR **6913** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VOBS_NO_VTSI
VTS <number> VTSTT_VOBS <file name> is present on disc, while the matching VTSI <file name> is not.

When the Title VOBS VTSTT_VOBS “VTS_0x_y.VOB” of one of the three possible VTSs is stored on disc, the matching VTSI “VTS_0x_0.IFO” must also be recorded.

[DVD+VR] ERROR **6914** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VTSI_NO_VTSI
VTS <number> VTSI <file name> is present on disc, while VTS <number> VTSI <file name> is not.

The VTSI file “VTS_0x_0.IFO” of VTS x can only be recorded on disc when the VTSI of all lower numbered VTSs is also recorded.

[DVD+VR] ERROR **6916** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VIDEO_TS_DIR_ILL
The VIDEO_TS directory contains file <name>, which shall not be stored there.

A file other than the ones allowed (VMGI, VTSI and VOBS files) is stored in the VIDEO_TS directory.

[DVD+VR] ERROR **6917** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_VOBS_NO_VOBS
VTS <number> VTSTT_VOBS <name> is present on disc, while VTS <number> VTSTT_VOBS <name> is not.

For all recorded VTSes, the same number of VTSTT_VOBS files must be recorded and registered in the file system.

[DVD+VR] ERROR **6918** (ref. [DVD+VR] 2.4.2)
ERR_DVDVR_FILE_SIZE_1
File <name> may not have been recorded properly ! It has a size of 1 byte, while its BUP version is <value> bytes.

When IFO and BUP file sizes are different while the files are required to be identical, one of the files presumably is corrupt, probably because something went wrong during recording (e.g. “bad spot write error”).

- Currently this is only considered a recording problem reported by this message if the ‘corrupt’ file has a file size of 1 byte specified by the file system(s).

[DVD+VR] ERROR **6919** (ref. [DVD+VR] 2.3.7)
ERR_DVDVR_FILE_SIZE_DIFF
IFO and BUP versions of the <VRMI | VMGI | VTSI> file must be bit-true copies, But <IFO file name"> has a size of <value> bytes but <BUP file name> is <value> bytes.

The size of the original and backup versions of the specified navigation/recording data file is not equal as required.

[DVD+VR] ERROR **6920** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_FILE_BOUNDARY
All files with reserved names shall start at a 32 kB boundary; But <file name> starts at sector <value> (<hex value>).

One of the DVD+RW Video specific files (which have a reserved pre-defined file name) is not 32kByte aligned.

[DVD+VR] ERROR **6921** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VTSTT_VOBS_LOC_DIF
All VTSTT_VOBS files must have the same logical sector start address, but <file name> starts at sector <value>, instead of <value>.

[DVD+VR] ERROR **6922** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VTSTT_VOBS_SIZE_DIF
All VTSTT_VOBS files must have the same size, but <file name> has size <value> bytes, while <file name> size <value> bytes.

[DVD+VR] ERROR **6924** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VTSTT_VOBS_LOC
All VTSTT_VOBS files must start at logical sector start address <hex value> (<value>), but <file name> starts at sector <hex value> (<value>).

All VTSTT_VOBS files have a fixed start address: sector 0x4000.

[DVD+VR] ERROR **6925** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VTSTT_VOBS_SIZE
The total VTSTT_VOBS size shall be maximum <value> sectors, but the sum of all VTS <number> title VOBS files ' VTS_01_*.VOB ' is <value> sectors, ending at sector <hex value>.

The maximum total VTSTT_VOBS file size for title VTS x, which consists of the concatenation of all VOBS files “VTS_0x_y.VOB”, is limited to 2277184 sectors (4.7 GB).

- This is only verified for the first Title (VTS 1) only, since the VOBS files of other Titles have already been checked to be of equal size of the matching first Title file.

[DVD+VR] ERROR **6926** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VTSTT_VOBS_CONT
VTSTT_VOBS files shall be allocated contiguously, but <file name> ends at sector <value>, and <file name> starts at sector <value>.

There shall be no gaps (unrecorded sectors) between each two VOBS files of a Title VOBS VTSTT_VOBS.

- This is only verified for the first Title (VTS 1) only, since the VOBS files of other Titles have already been checked to be of equal size of the matching first Title file.

[DVD+VR] ERROR **6931** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VRM_SCR_NOT_1ST
VRM Scratch area is located at sector <value>. It must be the first data on the disc, but <file name> is stored at sector <value>.

The VRM Scratch area file is not recorded immediately after the File System data and before all other data files.

[DVD+VR] ERROR **6932** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VRM_USER_LOC
VRM User Data file <file name> starts at sector <value> which locates it < in the DVD-Video Zone | before the VRMI data>.

When VRM User Data files are recorded, they must be located between the VRMI data file and the first DVD-Video Zone data file, i.e. VIDEO_TS.IFO.

[DVD+VR] ERROR **6933** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_FILE_ORDER
The files shall be allocated in a fixed order; But <file name> is located at sector <value> which is not after <file name>, starting at sector <value>.

The specified file is not recorded according to the prescribed order.

[DVD+VR] ERROR **6935** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_USB_LENGTH_MIS
Incorrect Unallocated Space Bitmap Length (<value> bytes) specified
in Partition Descriptor of UDF file system when discsize is <8|12>cm.
The correct size for USB Length must be <value> bytes.
This is a cross check between Lead-In control data zone – Physical Format Information (disc size) and the size of the USB of UDF file system.

[DVD+VR] ERROR **6936** (ref. [DVD+VR] 2.4.4.3)
ERR_DVDVR_PD_FIELD_MIS
Because the disc contains a Data Section (VRMI_GI.VR_APP=01h),
the UDF Partition Descriptor field '<string>' must be <value>,
but currently is <value>.
This is a cross check between VRMI and UDF file system.

[DVD+VR] ERROR **6937** (ref. [DVD+VR] 2.4.4.3)
ERR_DVDVR_FLAG_NOT0
Because the disc contains a Data Section (VRMI_GI.VR_APP=01h),
the UDF Partition Descriptor field '<string>' must be 0.
This is a cross check between VRMI and UDF file system.

[DVD+VR] ERROR **6938** (ref. [DVD+VR] 2.4.4.3)
ERR_DVDVR_USB_POSITION_ILL
The Unallocated Space Bitmap (USB) is not fully recorded within the Data Section.
USB start address is at logical sector <value>,
while the Data Section starts after logical sector <value>.
This is a cross check between VRMI and UDF file system.

[DVD+VR] ERROR **6939** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VTSM_VOBS
VTS <number> VTSM_VOBS file <file name> is present on disc, but no VTS menu VOBS are allowed.

No VTS Menu VOBS are allowed in DVD+VR.

[DVD+VR] ERROR **6945** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_VRMI_INVALID
File <"VIDEO_RM.IFO" | "VIDEO_RM.BUP"> has probably not properly been recorded ! It has a size of <value> bytes while exactly <value> bytes are required. It is invalidated <and the backup data is used instead | and its data is not used for cross checks>.

This message is generated when one of the VRMI data files has been found to be invalid, reporting this fact to the user. This currently is the case when its file size is not equal to the required size. In case of the IFO file it reports to revert to the use of the BUP files; In case the BUP file is found to be invalid too, it reports no cross checks data will be available.

[DVD+VR] ERROR **6946** (ref. [DVD+VR] 3.3.2-3)
ERR_DVDVR_IFOBUP_INVALID
File <"VIDEO_TS.IFO" | "VIDEO_TS.BUP" | “VTS_0x_1.IFO” | “VTS_0x_1.BUP”> has probably not properly been recorded ! Its size is only <value> bytes. At least <value> (<value> sectors) are required. It is invalidated <and the backup data is used instead | and its data is not used for cross checks>.

This message is generated when one of the navigation data (VMGI or VTSI) files has been found to be invalid, reporting this fact to the user. This currently is the case when its file size is not equal to the required size. In case of the IFO file it reports to revert to the use of the BUP files; In case the BUP file is found to be invalid too, it reports no cross checks data will be available.

9.4.10.2 File Systems Specific Checks

These checks relate to the UDF or ISO-9660 file system itself.

ISO9660 Checks

[DVD+VR] ERROR **6951** (ref. [DVD+VR] 2.4.3)
ERR_DVDVR_FS_MULTI_EXTENT
All files shall be recorded with a single extent, but <file name> has <value> extents.

Every extent of each file defined by the file system is inspected for being not a final extent. This indicates that a specific file consists of more than one extent, which is not allowed in DVD+RW Video.

- This is actually only verified for the ISO-9660 file system.

[DVD+VR] ERROR **6955** (ref. [DVD+VR] 2.2.4)
ERR_ISO_CMGS_NOT0
The ISO-9660 file system specifies non zero CGMS data: <field name> field is <hex value>.

All CGMS fields in the file system data must be set to zero. More specifically this applies to:
- GMS Information CGMS
- CGMS Information Copyrighted Material
- Protection System Information Protection System Type

UDF Checks

[DVD+VR] ERROR **6956** (ref. [DVD+VR] 2.2.4)
ERR_UDF_CMGS_NOT0
The UDF file system specifies non zero CGMS data: <field name> field is <hex value>.

All CGMS fields in the file system data must be set to zero. More specifically this applies to:
- GMS Information CGMS
- CGMS Information Copyrighted Material
- Protection System Information Protection System Type

[DVD+VR] ERROR **6958** (ref. [DVD+VR] 2.4.3)
ERR_UDF_VOBS_ICB_DIFF
The UDF file system File Identifier Descriptor for file <name> specifies its ICB at logical block <value>. This must be identical to the ICB address <value> of file <name>.

To assure that VTS files “VTS_0x_y.VOB” with the same value for y have the same contiguous logical space allocated, they must have the same ICB.

Disabled DVD Checks

None.

9.4.11 Cross Checks

9.4.11.1 VOBU Cross Checks

[DVD+VR] WARNING **6961** (ref. [DVD+VR] 3.3.4.1)
ERR_DVDVR_ XCHECK_VOBU_AST_NOT_AVAIL
This VOBU part of a Cell referred to by PGC <number> has no audio packet of decoding stream <number>, although its Availability flag is set.

The Availability flag value for each possible audio stream in the VOBU is retrieved via the current C_IDN value from its corresponding PGC, while the presence of one of the 8 possible audio streams has been marked upon an (audio) PES_packet_header event. As such it cross-checks between the VTSI – PGCI data and the actual VOBS – MPEG data.

- This is only coded as a warning since this requirement may be removed in the future.

[DVD+VR] INFO **6962** (ref. [DVD+VR] 3.3.3.7)
ERR_DVDVR_XCHECK_VOBU_SA_NOT_PRESENTED
The VOBU with Start Address <value> is not present in the VOBS stream although it is recorded in the VTS_VOBU_ADMAP table.

This is checked at the end of a verification run, reporting all the VOBUs listed in the VTSI VOBU address table that have not been encountered in the VOBS data. As such it is a cross check between VTSI and VOBS.

Note that in DVD+RW Video and unlike the VTSI Cell address table (VTS_C_ADT) containing only the data for the Cells that are really part of the matching VTS, the VOBU address table holds the data of all VOBUs of all VTSs on the disc!

Since this is strictly not incorrect, it is reported as an information message.

[DVD+VR] ERROR **6963** (ref. [DVD+VR] 4.3(32-0))
ERR_DVDVR_XCHECK_VOBU_SIZE_MISMATCH
The VOBU_SIZE <value> in VRMI_CHPI's VRCHP_IT[<number>] is different from the number of packs <value> in the VOBU at address <value> (Cell ID <number>).

This is an inconsistency between the VRMI and VOBS data, i.c. size of the VOBU.

[DVD+VR] ERROR **6964** (ref. [DVD+VR] 4.3(32-0))
ERR_DVDVR_XCHECK_CELL_START_MISMATCH
The VOBU at address <value> is not the first of the Cell with ID <number> although a Cell start flag <value> in VRMI_CHPI VRCHP_IT[<number>] indicates it as the first VOBU of that Cell.

This is an inconsistency between the VRMI and VOBS data, i.e. VRMI incorrectly flags a VOBU as being the start of a Cell.

9.4.11.2 Bit rate Cross Checks

The first 3 checks are done at the end of each pack (at an EVT_PACK_HEADER event), while the next is done at the end of each VOBU (at an EVT_VOBU_END event).

[DVD+VR] ERROR **6965** (ref. [DVD+VR] B.1)
ERR_DVDVR_XCHECK_VOBU_NOT_CBR_HQ
The SCR difference <value> is not equal to <value> as required for constant Bitrate in High Quality mode for recording <number>.

Constant bitrate recording (CBR) in High Quality mode demands a constant difference (= 45530 ticks) between consecutive SCR values.

[DVD+VR] ERROR **6966** (ref. [DVD+VR] B.1)
ERR_DVDVR_XCHECK_VOBU_NOT_CBR
The SCR difference <value> is not a multiple of <value> as required for constant Bitrate of recording <number>.

Constant bitrate recording (CBR) in non High Quality mode demands a fixed relation between the difference of consecutive SCR values and the Bitrate level (with factor 10900).

$$\Delta SCR = BL \times 10900$$

[DVD+VR] ERROR **6967** (ref. [DVD+VR] B.1)
ERR_DVDVR_XCHECK_VOBU_NOT_CBR_BL
The SCR difference <value> divided by <value> does not match the Bitrate level <value> of recording <number>.

Constant bitrate recording (CBR) in non High Quality mode demands a fixed value for the difference of consecutive SCR values which is only dependent on the Bitrate level :

$$\Delta SCR = BL \times 10900$$

[DVD+VR] ERROR **6968** (ref. [DVD+VR] B.2)
ERR_DVDVR_XCHECK_VOBU_NOT_CVBR
The Playback time <value> computed for recording <number>'s piece between the <VOBU | VOB start> at LBN <value> and VOBU ending at <value>, deviates more than the allowed (37.5 * <value> (BL)) seconds from the CBR playing time for that Bitrate level <value>, which is required for < CVBR | Mixture of CBR and CVBR>.

Constrained variable bitrate recording demands:

$$\left|Playback\_Time - CBR\_Time\right| \leq 37.5 \times BL$$

This is actually checked twice:

At the end of each VOBU, with:

$$Playback\_Time = VET_{current} - VST_{current}$$
$$CBR\_Time = VEB_{current} - VSB_{current}$$

and

$VST_{current} = VOBU\_S\_PTM$ of the current VOBU
$VET_{current} = VOBU\_E\_PTM$ of the current VOBU
$VSB_{current} =$ LBN of the current VOBU's NV_PCK
$VEB_{current} =$ LBN of the current VOBU's last pack

At the end of a VOB, with:

$$Playback\_Time = VET_{last} - VST_{first}$$
$$CBR\_Time = VEB_{last} - VSB_{first}$$

and

$VST_{first} = VOBU\_S\_PTM$ of the VOB's first VOBU
$VET_{last} = VOBU\_E\_PTM$ of the VOB's last VOBU
$VSB_{first} =$ LBN of the VOB's first NV_PCK
$VEB_{last} =$ LBN of the VOB's last pack

! This is actually also checked in case a mixture of CBR and CVBR is specified, since CVBR is considered to be less restrictive than CBR.

[DVD+VR] ERROR **6969** (ref. [DVD+VR] B.3)
ERR_DVDVR_XCHECK_VOBU_NOT_MIXED_BR
The Bitrate for recording <number> does not fulfil the requirements for CBR or for CVBR, as required for mixed CBR and CVBR mode.

This is currently not implemented, since it is partly covered by the previous check. Whether some parts of a recording are actually CBR encoded is hard and not very useful to verify.

9.4.11.3 VRMI Cross Checks

The following checks are verify the consistency between VRMI and other navigation data (VMGI, VTSI, PGCI).

VMGI & VRMI

[DVD+VR] RECOMMENDATION VIOLATION **6971** (ref. [DVD+VR] 3.2.4)
ERR_DVDVR_XCHECK_VMGI_CONSECUTIVE_FTIT_DELETED
Two consecutive full titles shall never be tagged as deleted. The titles <value> and <value> are tagged as deleted.

Two consecutive full titles shall never tagged as deleted. If a recording is deleted both full title & play list title have a unique playback type value in TT_SRPT (TT_PB_TY UOP1 must be '1').

[DVD+VR] ERROR **6972** (ref. [DVD+VR] 3.2.6)
ERR_DVDVR_XCHECK_VMGI_NOADDL_TITLE_FREE_SPACE
The disc is <not | > full, so the last <Play List | Full> Title TT_SRP[<index>] must be a <Free Space | Real> Title, but it is a <Real | Free Space> Title.

One additional Play list and one additional Full title tagged as free space must be added to TT_SRPT when the last recording is not a deleted recording (if number of recordings is smaller than 49 and there is at least 4MB space on disc).

[DVD+VR] ERROR **6973** (ref. [DVD+VR] 4.2(64))
ERR_DVDVR_XCHECK_VMGI_TVSYSTEM_NOT_IDENTICAL
VRMI's Disc TV System <value> must be identical to the TV system <value> as specified in VMGI and VTSI.

Disc TV system in VRMI is identical to TV system specified in VMGI.

[DVD+VR] ERROR **6974** (ref. [DVD+VR] 3.3.2.2, 4.2(1024))
ERR_DVDVR_XCHECK_VMGI_NUM_FTT_MISMATCH
Number of full titles in the VRMI <value> and number of full titles found in TT_SRPT <value> do not match.

The number of VRMI FTT_Ns and number of TT_SRPT recorded in the VMGI must be equal.

[DVD+VR] ERROR **6975** (ref. [DVD+VR] 3.2.6)
ERR_DVDVR_XCHECK_VMGI_DELREC_NOT_FREESPACE
The recording in VRMI_RECI[<index>] has been deleted, but the corresponding play list title TT_SRP[<index>] & full title TT_SRP[<index>] in VMGI is not tagged as Free space.

For every deleted recording the corresponding full title & play list title must be tagged as free space.

[DVD+VR] ERROR **6976** (ref. [DVD+VR] 4.4.1)
ERR_DVDVR_XCHECK_VMGI_FULLTITLE_ORDER_ILLEGAL
The recorded blocks in VRMI_RECI are not in the same order as the associated full titles in TT_SRPT in VMGI.

Recording information blocks must be in the same order as associated full titles in TT_SRPT.

[DVD+VR] ERROR **6977** (ref. [DVD+VR] 3.2.6)
ERR_DVDVR_XCHECK_VMGI_RECTIT_MISMATCH
The number of recordings (<value>) does not match number of titles (<value>) found. The total number of recordings (both existing and deleted) must be one less than total number of full title specified in VMGI.

VTSI & VRMI

[DVD+VR] ERROR **6978** (ref. [DVD+VR] Annex D)
ERR_DVDVR_XCHECK_VTSI_xTOT_LARGE
The total number of '<value>' for all '<value>' Titles (<value>) is larger than the allowed <value>.

[DVD+VR] ERROR **6979** (ref. [DVD+VR] Annex D)
ERR_DVDVR_XCHECK_VTSI_xTITLE_PGS_ILL
The number of programs (<value>) in Play List Title (Title <value>) must be less than <value>, twice the number of programs (<value>) in the related Full Title (Title <value>).

[DVD+VR] ERROR **6980** (ref. [DVD+VR] 3.3.3.7, 4.41(0), 4.2(1024))
ERR_DVDVR_XCHECK_VTSI_VOB_SA_ILLEGAL
VOBU start address <value> of a deleted recording VRMI_RECI[<index>] is present in the VTS_VOBU_ADMAP which is not allowed.

VOBU start address in VTS_VOBU_ADMAP shall not contain the deleted recordings.

[DVD+VR] ERROR **6981** (ref. [DVD+VR] 3.2.4, 4.4.4(28))
ERR_DVDVR_XCHECK_VTSI_LAST_DELREC_VOBSA_FOUND
VOBU start address <value> of last deleted recording VRMI_RECI[<value>] is present in the VTS_VOBU_ADMAP which is not allowed because VOBUs that are part of deleted recordings are not included in VTS_VOBU_ADMAP.

VOBUs of deleted recordings shall not be included in the VTS_VOBU_ADMAP.

[DVD+VR] ERROR **6982** (ref. [DVD+VR] 3.3.3.7, 4.3(0))
ERR_DVDVR_XCHECK_VTSI_VOBU_A_DOES_NOT_EXIST
VOBU address <value> of VRCHP_IT[<value>] is not found in the VTS_VOBU_ADMAP table.

All VOBUs included in existing recordings must be present in the VTS_VOBU_ADMAP table.

[DVD+VR] ERROR **6985** (ref. [DVD+VR] 4.4.3(1))
ERR_DVDVR_XCHECK_VTSI_VRPL_PL_MATCH_NR_ERR
The number of VRPL entries <value> for recording VRMI_RECI[<index>] is different from the number of Programs <value> in the matching Play List Title. However PL match is <value>, specifying PL match is <VOBU | vframe> accurate.

Since the PL match flag specifies VOBU or video frame accuracy between the Play List Title and the VR Play List, at least the number of chapters in both should match.

[DVD+VR] ERROR **6986** (ref. [DVD+VR] 4.4.3(1))
ERR_DVDVR_XCHECK_VTSI_VRPL_PL_MATCH_ADR_ERR
VRMI_RECI[<index>] VRPL entry <number> refers to chapter <number> in VRMI_CHPI with marker at VOBU address <value>.
Although PL match is <value>, specifying PL match is <VOBU | vframe> accurate, there is no Program defined in the DVD navigation data with its first Cell starting at the same location (Program <number> starting at <value>; Program <number> at <value>)

In case of video frame accuracy, the VRMI_CHPI CHP_VOBU_A start address matching any of the VR Play List VRPL entries, is compared with any of the DVD Play List Title PG start address C_FVOBU_SA to see if there is a pair matching exactly, as required.

The case of VOBU accuracy has not been implemented yet. Being much harder to implement and actually a cross check between VRMI and VOBS data.

[DVD+VR] ERROR **6987** (ref. [DVD+VR] 4.3(32-0))
ERR_DVDVR_XCHECK_VTSI_VOBU_A_NO_CELL_START
VRMI_CHPI: The VRCHP_IT[<index>] VRCHP_MKI CHP_VOBU_A is <value>.
However this does not mark the start of the first VOBU of a Cell which is required since the <Cell | Rec> start flag is set.

When either the Rec start or Cell start flag has been set, the matching VRCHP_MKI CHP_VOBU_A address value is looked-up in the VTSI VTS_C_ADT Cell address table to check if it is indeed a valid Cell start address.

[DVD+VR] ERROR **6988** (ref. [DVD+VR] 4.2)
ERR_DVDVR_XCHECK_VCPS_ID_CP_METHOD
VCPS_ID in the <VMGI/VTSI> indicates that this is a VCPS disc with an Extended <VMGI/VTSI> Structure for Supplementary Navigation. If Supplementary Navigation is used, the disc shall have VRMI_GI: CP_Method = 02h.

This error indicates supplementary navigation is used in the VMGI or VTSI but CP_method is not 02h.

[DVD+VR] ERROR **6989** (ref. [DVD+VR] 4.2)
ERR_DVDVR_XCHECK_VRMI_GI_VERN_VCPS_ID
VCPS_ID in the <VMGI/VTSI indicates that this is a VCPS disc with an Extended <VMGI/VTSI> Structure for Supplementary Navigation. If Supplementary Navigation is used, the disc shall have version number VRMI_GI: VERN = 0030h.

This error indicates supplementary navigation is used in the VMGI or VTSI but is not 0030h

[DVD+VR] RECOMMENDATION VIOLATION **6990** (ref. [DVD+VR] Annex D)
ERR_DVDVR_XCHECK_TOO_MANY_CELLS
The recommended amount of used, reserved and planned Cell IDs exceeds the maximum amount of available Cells. Cused (<value>) + Cres (<value>) + Cplan (<value>) should be less than <value>.

This is a recommendation violation message.

PGCI & VRMI

[DVD+VR] ERROR **6991** (ref. [DVD+VR] 4.4.3(16))
ERR_DVDVR_XCHECK_PGCI_PB_TM_NOT_IDENTICAL
The playback time of the full title as specified in VRMI_RECI[<index>] <playback value> is not identical to the playback time as specified in PGCI with PGCN <value> <playback value>.

REC_PB_TM specifies playing time for full title.

[DVD+VR] ERROR **6992** (ref. [DVD+VR] 4.4.3(1))
ERR_DVDVR_XCHECK_PGCI_FPTITLE_NOT_IDENTICAL
All programs of the Play list title with PGCN <value> and full title with PGCN <value> are not identical although the PG match <value> of VRMI_RECI<value>] is set.

PG match in VRMI_RECI is set if and only if (iff) PGC_PMAP for PGCI for play list title are identical to programs defined in PGC_PGMAP of PGCI for full title. (These are identical when related sequences of C_POSI blocks in two C_POSIT are identical).

[DVD+VR] ERROR **6993** (ref. [DVD+VR] 4.4.3(1))
ERR_DVDVR_XCHECK_PGCI_FULLTITLE_NOT_FOUND
The corresponding program chain for VRMI_RECI[<index>] full title was not found.

Corresponding recorded full title does not have a corresponding program chain.

[DVD+VR] ERROR **6994** (ref. DVD+VR 4.4.3(16))
ERR_DVDVR_XCHECK_PGCI_TC_FLAG_NOT_IDENTICAL
The tc_flag of the REC_PB_TM of the Full Title as specified in VRMI_RECI[<value>] (<value> fps) is not identical to the tc_flag as specified in PGCI with PGCN <value> (<value> fps).

The tc_flag specified in VRMI does not correspond with tc_flag specified in PGCI.

[DVD+VR] INFORMATION **6995** (ref. [DVD+VR] 3.2.6)
ERR_DVDVR_XCHECK_VMGI_ADDL_FS_INFO
The disc is considered full because <The number of recordings is | The space left for a new recording is only> <value> <which is the maximum number allowed | but at least 4 MB is required>.

This is merely an informative message notifying the user that the verifier considers this disc as being full, because of the specified reason.

[DVD+VR] ERROR **6996** (ref. [DVD+VR] 3.2.6)
ERR_DVDVR_XCHECK_VMGI_ADDL_FS_CONDN
No additional free space play title and free space full title found in TT_SPRT. For one additional Play list title and one additional full title tagged as free space to be added to TT_SRPT in VMGI, the number of recordings (<value>) must be lesser than 49 and the space for a new recording (<value>) must be at least 4 MB.

[DVD+VR] ERROR **6998** (ref. [DVD+VR] 2.2.4)

ERR_DVDVR_XCHECK_FS_SECT_CGMS

The UDF File System CGMS bits are <value> for file <file name> but the sector CGMS bits are not <0 | 1> in <all of its sectors | at least one sector>.

This is a cross check between the CGMS bits as recorded in the file systems(s) and the value set in the header of all sectors of this file.

It is only checked for the UDF file system. Possible inconsistencies with the ISO-9660 file system will be reported by the "inter file system" cross checks as differences between both file systems.

9.4.11.4 Content Protection

[DVD+VR] ERROR **8000** (ref. [DVD+VR] 4.2(65))

ERR_DVDVR_CP_VRMI_GI_METHODE_RESERVED

CP_METHOD in the VRMI_GI indicates that this is a VCPS encrypted disc. If this is a VCPS encrypted disc the disc shall have version number VRMI_GI: VERN = 0020h or 0030h.

10 VERIFIER USE AND BEHAVIOUR NOTES

This section describes the format of messages generated by the DVD+RW Video Format Verifier and the output as a result of the dump options.

10.1 ADVISE

If the verifier input data has a serious deviation with respect to the supported standards, a possible crash during the parsing phase might occur. In that case our advise is to fix the already reported errors first and then try again, since the crash is likely to be a direct result of these earlier reported errors.

10.2 GUIDELINES FOR USE

1. Use a verification start offset with extreme caution, only when really necessary and always specify a Cell start location to avoid “run-in” problems.
2. Unless there are very good reasons to do otherwise, skip parsing for only those levels that result in a significant decrease of the verification time.
3. Only if really necessary, verification of certain levels should be skipped, in order to avoid side effects such as unjustified error messages.

10.3 TIPS

- A significant speed-up of the verification process is obtained when skipping the MPEG Video macroblock parsing (through the “-**SM**” command-line option or by checking the GUI ‘Skip decoding’ ‘Macroblock’ button).
- Additionally, if the disc has Dolby AC-3 audio encoded, verification time is shortened by skipping AC-3 decoding (through the “-**Sa**” command-line option or by checking the GUI ‘Skip decoding’ ‘Dolby AC-3 audio button).
- Skipping the Lead-in (through the “-**S0**” command-line option or by checking the GUI ‘Skip decoding’ ‘Lead-in’ button), which implies effectively skipping Lead-out too (since it is no longer known if it is a temporary Lead-out or not and where it is located), also contributes to accelerated verification. Certainly if one is not interested in low-level verification. Anyhow, there is no danger in doing it, since the really indispensable information is always read from the Lead-in.
- When skipping the first part of a stream (through the “-**f***N*” command-line option or by specifying a start position in the ‘Misc settings’ ‘Start verification at” box), one should specify at least the start position of a VOBU (as encoded in the VTSI VOBU_ADMAP table) and preferably the start of a Cell (as encoded in the VTSI VTS_C_ADT table), to avoid unjustified messages or other side effects.

11 VERIFIER IMPLEMENTATION SPECIFICS

11.1 VTSI CELL DATA CONTROLLED PARSING

DVD+RW Video discs can have up to and at most 3 VTS title VOBS. Another characteristic is that all VTS_TT_VOBS present on a disc exactly and completely overlap. This means that in a way their files contain ‘gaps’ at the locations of data that actually belongs to one of the other VTSs (cf. [DVD+VR]) or data that has been invalidated or erased. The matching VTSI data (as stored in VTS_0x_0.IFO) exactly describes (in the Cell address table VTS_C_ADT) which Cells are actually present in a VTS and at which RLBN locations.

That is exactly how the verifier parser accesses the VTS files for verification : It uses the VTSI recorded Cell address table to parse only the data sectors of the Cells that are actually part of the VTS VOBS.

As a consequence in case of multiple VTSs and if the VTSI data is missing or incorrect (which will be reported by the verifier), correct parsing of a VTS VOBS is not possible! If one decides to go ahead with verification, the verifier will simply parse ALL data, incl. uncorrelated data of another VTS or invalid garbage data, which is likely to result in unjustified error messages.

11.2 VR PLAY LIST CONTROLLED PARSING

As an alternative to the VTSI Cell data controlled parsing as described above, which is actually “Full Title” parsing per VTS, the VR Play List data (as specified in the VRMI) can be used to control the parsing process. This allows to take into account possible user edits in the parsing process and only verify those parts of the stream data that have not be skipped (erased or marked as such) by the user and possibly in the changed order as specified by the user.

This VR Play List is the recorder generated VOBU or video frame accurate “VR Play List” as encoded in the VRMI data.

It is not the DVD-Video compliant only Cell accurate “DVD Play List” as encoded in the VTSI / PGCI data (see previous section).

The VR Play List parsing is selectable with the “-**PV**” command-line option or through the GUI ‘Misc Settings’.

Since play list jumps are VOBU and may not be Cell boundary aligned, certain checks have to be temporarily disabled to avoid unjustified error messages:
e.g. error **6054** and **6959** until the next pack, and the DSI VOBU_SRI checks (**4680** .. **4687**) and PCI checks **4534** .. **4536** until the next Cell start.

11.3 SELECTIVE PARSING & VERIFICATION

Certain 'low' level data structures are not always parsed & verified. This is done, either to speed-up verification or to avoid repetition of identical error messages.

File system(s) data is always parsed since it specifies the actual contents of the disc(image) and through their location and size, how to access the different files. Verification however is only done (unless disabled by the user) in case either the complete disc(image) is being verified (default or "-**XA**" option), or when file system only verification has been specified ("-**Xf**" option).

Lead-in and **Lead-out** data (unless not present or disabled by the user) is only parsed and verified in case either the complete disc(image) is being verified (default or "-**XA**" option), or when lead-in resp. lead-out only verification has been specified ("-**XL**" resp. "-**XO**" option). The Lead-in PFI data sector is always parsed, regardless of user settings, to have always available the indispensable data fields it contains.

11.4 NAVIGATION FILE BACKUP VERIFICATION

All DVD+RW Video disc navigation data (VRMI, VMGI and VTSI) is stored twice on a disc: once in an "IFO" file ("VIDEO_RM.IFO", "VIDEO_TS.IFO" and "VTS_0x_0.IFO", x=1..3) and once in a backup "BUP" file ("VIDEO_RM.BUP", "VIDEO_TS. BUP" and "VTS_0x_0. BUP", x=1..3).

The backup has to be a bit-true copy of the original, incl. the relative addresses it contains.

The verifier handles this as follows:

1. Both files are parsed and verified separately. So also the backup is completely parsed and verified as if it were the original.
2. Both files are then sector-wise compared and if one of the sectors is different in both copies of the file, it is reported as an error.

In case both copies of a file are different, a contents dump may show the differences in detail.

11.5 CROSS CHECKING

The order in which files are processed is important for cross checking, since data needed for cross checks must have been stored in the cross check data file during earlier verification of other data files. So for complete and correct cross verification, the files have to be processed in the correct hierarchical order, being:

1. VRMI: not needing data from any other file
2. VMGI: only using some VRMI data for cross checking
3. VTSI: using both VRMI and VMGI cross data
4. VOBS: cross checks with data from all navigation files

Cross verification is performed in a slightly different manner in the GUI and command-line versions of the verifier.

In the **GUI version** of the verifier, cross checks are always automatically enabled. The necessary cross check data is stored on a file with a temporary name, which is always automatically deleted after use. Because this file is deleted at the end of a verifier run, no cross check data can be carried from one run to a next. It is necessary that all needed files are processed in the same verifier run! So e.g. processing the navigation files in one run, and processing the VOBS data in a next, will cause incorrect cross checking, since the navigation file data has been lost. Also, by skipping certain files for verification, it is possible that necessary cross check data is not generated and not present in the cross check data file. In both cases this will result in some verifier messages reporting this and certain cross checks not being performed. However, the correct order of processing is always guaranteed and there is no danger of inadvertently using cross check data from an earlier run on different disc data.

The **command-line version** of the verifier allows specifying an explicit name for the cross check data file (using the script file interface) and the generated file is preserved at the end of the verification. By default the verifier creates this file with the default name "dvd_verif_xdata.xcheck" in the current output directory, which will be overwritten by successive verifier runs. This makes it possible that incorrectly cross check data generated during a previous run on different disc(image) data is used, resulting in unjustified cross check error messages! Specifying a unique, disc(image) specific file name for this file may help to avoid this problem. One has to take care oneself to respect the required processing order of the files, so that all needed data is indeed present in the cross check file.

11.6 ORIGINAL VS. BACKUP (NAVIGATION) FILE USE

In principle, the original ("IFO") version and its backup ("BUP") of a DVD navigation data file (i.e. VMGI and VTSI) or the DVD+RW recording data file (with the VRMI), are bit-true copies of one another. So it does not matter which data is stored in the cross check data file to perform cross checking with other data on the disc. However it does matter in case either the IFO or BUP file is unreliable or corrupt. Although the verifier by default uses the IFO data, the following strategy is implemented to deal with corrupted file data.

It relies on the assumption that if during the recording process a (write) problem has occurred not allowing to properly update (i.e. rewrite) such a file, it is described by the file system as having an incorrect, i.c. too small, size, thus marking it as invalid. [DVD+VR] specifies the file size must be set to 1 byte in this case.

If the verifier detects an IFO file marked as invalid this way, it will automatically revert to the BUP file and store the backup data of the latter rather than the original IFO data to the cross check data file. This is also extended to IFO files with a size lower than allowed and to the BUP file: if the latter is marked as invalid too, none of the files will be used and so cross checking against its data will effectively be disabled. Of course, all of these non-default actions are properly reported by the verifier.

In the future, read I/O problems while accessing an IFO or BUP file also might cause the invalidation of that file.

Optionally the user may force the use of either the IFO or BUP file data by specifying the command-line option "-**U**<**IB**>" or through the GUI 'Misc Settings'. Then the fact that a file is 'marked' as invalid is ignored or a correct IFO can be discarded to use its backup version.

11.7 VOB, CELL, VOBU BOUNDARY DETECTION

Because missing the start or end of some basic VOBS data structure may have a serious impact on parsing and verification of DVD+RW VOBS data, the method used to detect these boundaries is included in this user manual.

Unlike DVD-Video, in DVD+RW Video start and end of these basic VOBS data structures not always coincide, e.g. there could be garbage or data of another VTS in between successive VOBUs, Cells or VOBs of one VTS_TT_VOBS.

Furthermore these are no longer as easy distinguishable as in DVD-Video and locating their boundaries not straightforward: e.g. all VOBs have the same ID number '1' (VOB_IDN) and Cell ID (C_IDN) numbers no longer have to be sorted; Cell IDs may even be re-used as a Buffer Cell ID!

To cope with this, the DVD+RW Video Format Verifier has some built-in boundary detection functionality to locate start and end of VOB, Cell and VOB. The criteria used are described here.

In almost all cases the detection relies on the availability of correct VTSI derived (cross check) data, since it uses the VOBU address table (VOBU_ADMAP) or Cell address table (VTS_C_ADT) data. As a consequence, boundary detection fails if no correct (VTSI) cross check data is available! In some cases there is a fall back option, which is also described here.

Note: Unless explicitly stated otherwise, the following only relates to VTS Title VOBs data. In principle for Menu VOBS data (VMGM Domain), the 'old' DVD-Video detection scheme still holds.

11.7.1 Start Detection

VOBU
(Exactly as done for DVD-Video) A VOBU start is detected by, being at the start of an MPEG pack, reading ahead and detecting at the expected positions in the stream:

- an MPEG PS system_header start_code
- a private_stream_2 PES_packet start_code
- a PCI sub-stream ID

Cell
Being at the start of a VOBU:

- At a Cell start when the current pack number (RLBN) matches the VTS_C_ADT start address of the current Cell

Fall back, in case no correct VTSI cross check data is available, or for Menu VOBS :

- At a Cell start, if the current Cell ID is different from the previous one.

VOB
Being at the start of a VOBU:

- At a VOB start, if the end address of the previous Cell is more than 1 sector apart from to the start address of the current Cell, and thus there is a gap (with garbage or other VTS data) before the current Cell.

Fall back, in case no correct VTSI cross check is available data or the previous or current Cell entry can not be found in VTS_C_ADT:

- At a VOB start, if the current VOB presentation start time VOB_V_S_PTM is different from the previous one.

  ➔ However this is not 100% reliable, since it is possible (however fairly unlikely) that the VOB_V_S_PTM for successive VOBs is identical, which is allowed.

For Menu VOBS data, the old DVD-Video approach is maintained:

- At a VOB start, if the current VOB ID is different from the previous one.

11.7.2 End Detection

If the start of any of these structures is detected, but the end of the previous occurrence has been missed, the latter will be notified just before sending out a notification of the start of the new instance.
At the end of a verification run, a notification for the end of any of these (VOBU, Cell, VOB) structures is (forcedly) generated.
Being at the end of an MPEG pack:

VOBU

- At a VOBU end, if the current pack number (RLBN) is one less than the start of the next VOBU as recorded in the VOBU_ADMAP.

  Remark: This strategy makes it impossible to properly detect the very last VOBU of a disc, since there is no start address of a successor encoded in the VTSI VOBU_ADMAP.

Fall back, in case no correct VTSI cross check data is available or the current or next VOBU can not be found in the list:

- A VOBU end is assumed at least at every Cell or VOB end

Cell

- At a Cell end when the current pack number (RLBN) matches the VTS_C_ADT end address of the current Cell

Fall back, in case no correct VTSI cross check data is available, or for Menu VOBS :

- Start and end of a Cell are assumed to coincide

VOB

- At a VOB end, when at the end of a Cell there is no Cell recorded in the current VTS_C_ADT starting at the next RLBN, meaning there is a gap between successive Cells.

Fall back, in case no correct VTSI cross check data is available, or for Menu VOBS :

- If the next Cell ID is a Buffer Cell ID (C_IDN==255), which is not guaranteed to work since the use of the specific Buffer Cell ID 255 is not mandatory.
- OR else start and end of a VOB assumed to coincide

11.8 DISABLED CHECKS IN CASE OF MISSING STREAM START

When the actual start of a data stream (i.e. some sectors or VOBUs) is missing and so making it impossible or at least unreliable to verify some issues, the related checks will be disabled to avoid unjustified messages. This situation may occur as the result of the user specifying a verification start position within the data stream, skipping the stream's start to analyse quickly a specific part of the data.

The disabled checks are listed here.

When the actual VOB start is missing:

VOBU alignment with video grid check

ERROR **4522** (ERR_DVD_PCI_VOBU_S_PTM_MULT)

CVBR bit rate check since the VOB start

ERROR **6979** (ERR_DVDVR_XCHECK_VOBU_NOT_CVBR)

When a Cell's start (at least 1 VOBU) is missing:

All DSI VOBU_SRI **backward** references checks:

ERROR **4671** (ERR_DVD_DSI_SRI_FBWD_EX_1)
ERROR **4673** (ERR_DVD_DSI_SRI_FBWD_EX_2)
ERROR **4675** (ERR_DVD_DSI_SRI_FBWD_EX_2_1)
ERROR **4678** (ERR_DVD_DSI_SRI_FBWD_EX_1_V)
ERROR **4679** (ERR_DVD_DSI_SRI_FBWD_EX_2_V)
ERROR **4680** (ERR_DVD_DSI_SRI_FBWDA_ILL)
ERROR **4681** (ERR_DVD_DSI_SRI_FBWDA_ERR)
ERROR **4682** (ERR_DVD_DSI_SRI_FBWDA_S_PTM)
ERROR **4684** (ERR_DVD_DSI_SRI_FBWDA_EXST)
ERROR **4686** (ERR_DVD_DSI_SRI_FBWDA_TIMEX)
ERROR **4687** (ERR_DVD_DSI_SRI_FBWDA_EXST_FLST)
ERROR **4689** (ERR_DVD_DSI_SRI_FBWD_NOPRED)

The PCI_GI C_ELTM validity checks:

ERROR **4534** (ERR_DVD_PCI_GI_C_ELTM_1ST)
ERROR **4535** (ERR_DVD_PCI_GI_C_ELTM)
WARNING **4536** (ERR_DVD_PCI_GI_C_ELTM_WARN)

12 DEFECTIVE MEDIA HANDLING

This section describes the way the verifier deals with defects on the data stream carriers, i.c. DVD+RW discs.

12.1 DVD+RW DISC BAD SPOTS

12.1.1 Problem Description

During the creation of DVD+RW (Video) discs, i.e. while actually recording data by means of a DVD+RW Video recorder or another device, it may happen that a disc proves to be unwritable on the current location because of a physical defect (“bad spot”) or signs of wear. Because typically the data has to be written in real-time, there is no possibility to do proper complex defect management. What is actually written to the disc in this case, how much of the real-time data that is lost and how this is attempted to be fixed so that any play back device can play the disc with at least acceptable artefacts, is very much recorder device dependent !

Furthermore, bad spot read errors (“BSRE”) when reading a disc with any play back device are more likely to occur than and seem to precede bad spot write errors (“BSWE”) encountered during recording.

12.1.2 Matching Verifier Behaviour

Dependent on the location on the disc, bad spots will show differently in the verifier output.

This is very much dependent on what measures the recorder has actually taken when experiencing write problems.

Remark: It is very likely that the write problems only occur when rewriting a disc for the n-th time. In this case, there might well be valid (i.e. perfectly conform the spec) but outdated data at the bad spot location. Since this makes the data only invalid w.r.t. the contents, this can of course not be detected by the verifier. However, it may be reported as cross check errors when the outdated bad spot data is not consistent with other newer rewritten data.

The currently implemented verifier behaviour when encountering bad spot write errors depends also on the data area of the disc and is described below:

12.1.2.1 Lead-in

12.1.2.1.1 FDCB

The FDCB must be written in the first possible DCB. But when recording at the normal location has failed, it may be located in any of the next (15 available) DCBs, until it has been written successfully.

This will be reported as a deviation from the spec by errors **6364** and/or **6365**.

12.1.2.1.2 PFI

192 (or at least 16) identical copies of the PFI data have to be recorded in the Lead-in Control Data zone. When a bad spot prevents successful recording of a PFI copy, it may be simply skipped, relying on successful copies further on. In principle the first readable copy is used during playback, so this may only have impact when the write error occurs when writing the first PFI copy.

As a result of a BSWE in the PFI, not all192 copies of the PFI data may be identical (or have only a different Data Zone end address from the 17th copy on), which will be reported by error **5016** or **5037**.

12.1.2.2 Lead-out

As a result of write errors when recording a Temporary Lead-out (in the Data Zone), some ECC blocks may not be written. This may cause it to appear smaller than the minimal size of 64 ECC blocks. This will then be reported by error **6425**.

12.1.2.3 File Systems data

Except for sectors 16 and 256, the location of all other file system data sectors can be chosen more or less at random. This will probably also occur when encountering write problems when recording the file systems data: try to record it in another ECC block.

Since this does not violate any specification requirement, it will not be reported by the verifier.

Only when severe problems have prevented to properly record/update one of both file systems making them inconsistent, this will show as cross check errors.

12.1.2.4 Navigation data

When write errors occur because of bad spots when trying to write or update one of the navigation (VMGI or VTSI) or recording (VRMI) data files, these files are flagged as invalid by specifying a size of 1 byte in the file systems (as specified by [DVD+VR] 2.4.2).

This will also be interpreted adequately by the verifier and reported by error **6918**, **6945** or **6946**.

12.1.2.5 AV data

12.1.2.5.1 Typical Bad Spot Related Error Messages

Since the verifier can to some extent be considered as a ‘special’ play back device, it will experience the following problems at these bad spot locations:

- At least some data may be lost causing timing checks to fail, reported as e.g. ERROR **1131** or the parser to loose sync, e.g. while decoding the MPEG data, reported as syntax errors such as ERROR **1106**.
- The DVD-ROM or verification drive may experience read problems reported as ERROR **4983** or **4984**
- The disc I/O API may report read problems as ERROR **5006**

Currently the verifier simply reports any problems it finds and tries to recover, resyncs the parser and continues verifying the remaining data. In most cases this works. However because some data has been lost, at least the dynamic behaviour has changed and e.g. some buffer under or overflow problems are likely to be reported. It is also possible that recovery is not successful which might even result in a verifier crash!

12.1.2.5.2 Verification Abortion

When a 'typical' and unlikely combination of errors has been detected and reported, the verifier will assume this is caused by a bad spot on the disc under test. As a consequence further verification is considered not really useful, and in order to avoid more unjustified errors which are simply generated because undefined data has been returned by the drive, program execution is terminated.

However this premature verification termination can be avoided by specifying the command-line option "-**g**" or through the GUI ‘Misc Settings’.

12.1.2.5.3 Alternative Parsing Control

One method to (attempt to) make bad spot write errors less visible during play back is to adapt the DVD and Virtual Play List data during recording, so that the bad spot areas are automatically skipped during play back.

In this case, Play List controlled parsing (cf. 11.2 VR Play List Controlled Parsing) will result in a comparable behaviour of the verifier. It will also skip the bad spot area and hence (most of) the unjustified error messages generated by these.

However, this will not help against bad spot read errors, since these only occur during play back.

12.1.2.5.4 Optional Future Behaviour

In principle the verifier, being a DVD+RW Video data verifier performing conformance checks of the data against the DVD+RW Video specification, does not have to take any recorder specific issues into consideration. A disc with a bad spot write problem recorded is simply not compliant with the spec!

Other options that might be implemented in the future are:

- Make it jump to the next Cell when encountering a drive read error while reading plain VOBS data
- Only issue a master reset after having detected a bad spot and continue verification from scratch afterwards. Of course the buffer verification may no longer report problems, but has become useless, since it no longer reflects the actual expected case. The problem here is how a bad spot can be detected unambiguously and find its exact location.

13 INSTALLATION ISSUES

The package may be delivered in different forms: either as a simple ZIP file set, or as InstallShield package containing a setup, or a self-installing version of the latter.

13.1 SETUP

The tool installs as most Windows based tools by running a setup application, explaining its actions and prompting the user if some input is needed.

13.2 UNINSTALL

To uninstall the tool go to the Control Panel -> Add/Remove programs.

13.3 UNZIP

When the tool is available as a ZIP set, unpacking is straightforward (the unzip tool help can give additional information if needed).

14 KNOWN DEFICIENCIES

In this chapter a list of currently known problems or limitations with the DVD+RW Video Format Verifier is shown.

Bugs listed below will be solved in the near future. For the listed shortcomings and limitations there are no plans to do anything about them yet.

14.1 KNOWN BUGS

- <None known>

14.2 LIMITATIONS

- No generic ES or Private-stream input is supported.
- File system “on file”, i.e. not part of a complete disc (image), input is only supported for the command-line version of the verifier.
- Problems that are media related (e.g. "drop outs" or "bad spots" on an actual disc) can not handled adequately. It is also not within the scope of this tool to be able to handle it perfectly.
- Only DDP disc images using the (non-CMF) “0”-prefix file naming convention are supported by the GUI version of the verifier. The newer CMF format file names “DDVID.DAT”, “CONTROL.DAT” and “IMAGE.DAT” are only recognised as disc image files by the command-line version of the verifier.
- An error summary report for a complete disc (image) is only generated by the command-line version of the verifier.

14.3 SHORTCOMINGS

- Currently VRMI_RECI PL match is not checked yet in case it specifies “VOBU accuracy” (Error 6986).
- Verification start location can be specified in units not matching the input stream type and matching unsupported input stream types (non IFO or VOB files, i.c. PES or ES).
- Miscellaneous line width setting not (yet) active.

14.4 FEATURES NOT TESTED

Here, all the features, which were insufficiently - or not at all - tested, are listed. If something is not tested, this is usually the result of unavailable test data.

Consequently, a DVD+RW Video Format Verifier bug in data related to these aspects of the specification can not be excluded.

- <None known>

14.5 EXPECTED EXTENSIONS

- The DVD+RW Video Format Verifier will be extended in the near future to handle **IEC-60958 Audio** properly.

14.6 TROUBLESHOOTING

14.6.1 Verifier ERROR 5601, 4501 or 4601

If the cross check data file lacks some information needed to perform certain cross checks (because some of the navigation files containing cross check data has not been parsed (immediately) before or is not present at all (because it can not be located or has been deleted), one will get a **5601** ERROR message (and possibly **4501** & **4601** ERRORS as well) saying that, lacking a cross checks data file, the verifier will use some (possibly incorrect) default parameters or perform some checks not at all.

14.6.2 Incorrect Cross Check Data

Note that the command-line version of the verifier not always creates or updates a cross check data file, because e.g. only VOBS files are processed. It then uses any file with the expected name in the current output directory to retrieve its cross check data from. However this file may be the result of an earlier verifier run on a different disc (image) resulting in unjustified (cross check) error messages.

14.6.3 Disc Bad Spot Generated Errors

In case the verifier hits a “bad spot” (cf. **12.1 DVD+RW Disc Bad Spots**) on a DVD+RW disc where the write process has failed during recording, a waterfall of error messages may be expected. Indeed some data has been lost, timing data will be discontinuous, parsing will loose sync, buffer management will be disturbed, etc.

Typically at least some but in most cases all of the following error messages will be reported:

>>> [MPEG] SYNTAX ERROR **1106** :
Expecting packet_start_code etc. (Look Ahead : 0x……)

>>> [MPEG] ERROR **1131** (ref. MPEG Systems 2.4.5.2 | 2.7.1) :
Pack_header SCR difference is -2277.150 [90kHz ticks];
It shall be at most 0.7 seconds (= 63000.0 ticks).

>>> [MPEG] SYNTAX ERROR **1402** (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet has invalid timestamp mark (‘value’)

>>> [MPEG] SYNTAX ERROR **1424** (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet contains too many stuffing bytes (‘number’ > max ‘number’)

>>> [MPEG] SYNTAX ERROR **1425** (ref. MPEG Systems 2.4.4.3 | 2.4.3.7) :
PES_packet stuffing byte[‘index’] is ‘hexadecimal value’, should be 0xFF

>>> [APPL] SYSTEM ERROR **4983** (ref. ASA drive N/A) :
Drive I/O error : RETRY reading sector 1609729 (189001h) [LSN 1413121].

>>> [APPL] SYSTEM ERROR **4984** (ref. ASA drive N/A) :
Drive I/O error : retry reading sector 1609729 (189001h) [LSN 1413121] FAILED!

>>> [DSC API] ERROR **5006** (ref. Bigfile API) :
Could not read current sector (0x158EC1 / 1412801).

followed by lots of messages complaining about MPEG video & audio or AC-3 audio parsing (syntax errors) problems.

APPENDIX A PROBLEM REPORTS AND CHANGE REQUESTS

The following page contains a form that can be used to report problems, request changes or ask questions.

To get a quick reply always include the following information:

- A detailed description of the problem
- Urgency estimate of the problem
- The version number and release date of the verifier
- Command line arguments
- The input data, to be able to reproduce the reported problem
- A log file of the verifier output

CHANGE REQUEST / PROBLEM REPORT			
To be filled in by issuer		**To be filled-in by secretary CCB**	
Name issuer:		Name secretary CCB:	
Issue date CR/PR:		Type:	Change Request
Project:			Problem Report
Product / component:		CR/PR number:	
SW version / date:		Estimated Total effort:	
HW version / date:		Actual Total effort:	
Status	Owner	Date	Actual Effort
Registered			
Investigated			
Accepted			
Rejected			
Delayed			
Completed			
Approved			
Description:			
Remarks / Reason rejection:			

APPENDIX B VERIFICATION DRIVE

B.1 DESCRIPTION

To allow verification of ALL data on a DVD+RW disc a special drive has been developed by ASA Lab Eindhoven (as of January 1st, 2002, the departments name 'ASA-Lab' has been changed into PDSL – Philips Digital Systems Lab).

B.2 EXTRA DRIVE FUNCTIONALITY

When accessing a DVD+RW disc using this drive, the verifier tool offers additional functionality viz. the reading and verification of 'physical data' besides the logical data of the disc.
It's possible to:

- Read & verify Lead-in data mainly physical format Information (PFI) data
- byte sector header data and 4 byte 'tail'

Without the aid of this drive, it is not possible to either read or verify the Lead-in or sector data other than 2048 bytes the payload. Hence in the situation where this drive is not used, the verifier will have the following behaviour:

- If sector header data dump is requested, all zero sector header & tail data will be output.
- No checks will be done on sector header data
- Lead-in parsing is skipped by force
- Encrypted sector payload will be skipped.

This drive is not available outside Philips.
However to make use of the extra functionality, it is possible to send your disc to:

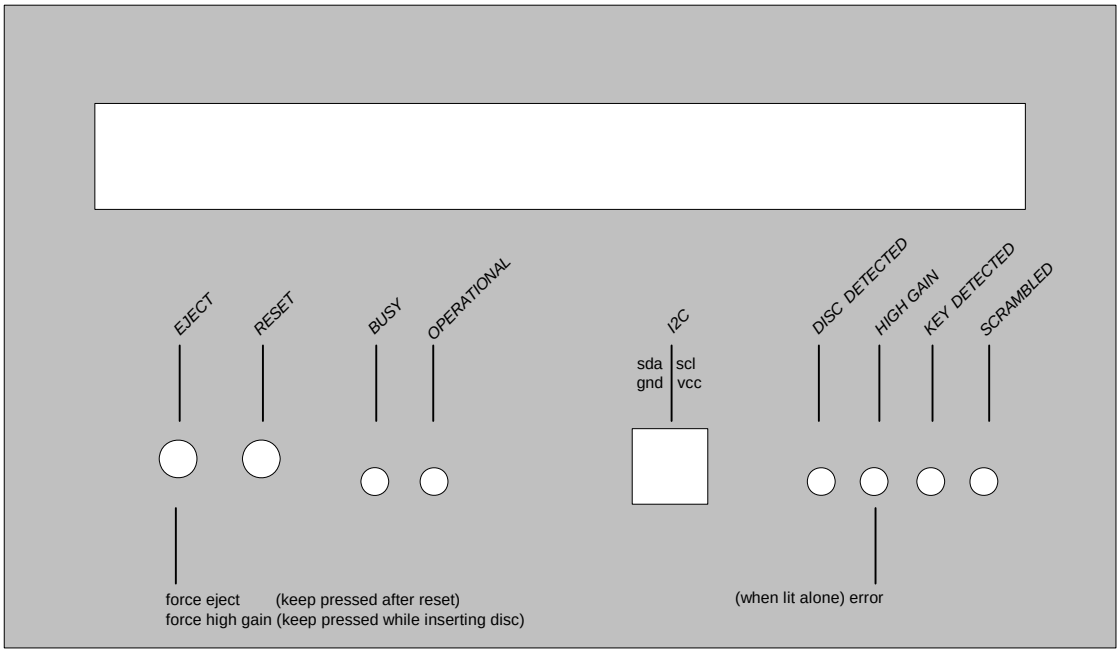
Mr. Maurice Hebben or Mr. Alan MacDuff,
Philips Digital Systems Lab – Eindhoven,
P.O. Box 80002
5600 JB Eindhoven
Building SFJ

B.3 INSTALLATION ISSUES

The drive interface to the host PC is through a standard IDE interface. However it must be configured as MASTER on it's IDE port.
The drive occupies a double height bay because of the extra PCB.

B.4 DRIVE USER MANUAL

FRONT PANEL



Push button: EJECT
Pressing this button while the tray is closed will open the tray. Pressing the button while the tray is open will close the tray.
Keeping the EJECT button pressed when resetting the drive will force the tray to open under any circumstance (sometimes the tray won't open when e.g. a disc is put in upside down):
1. Press the RESET and EJECT button at the same time
2. Release the RESET button
3. Wait until the HIGH GAIN / ERROR LED is active
4. Release the EJECT button
5. The tray will open

Keeping the EJECT button pressed while the tray is closing will force high gain mode (see the description of the HIGH GAIN LED):
1. Press the EJECT button to open the tray
2. Press the EJECT button to close the tray
3. Wait until the tray is closed and the HIGH GAIN / ERROR LED is active
4. Release the EJECT button
5. The drive is now in high gain mode

Push button: RESET
Pressing this button forces a hardware reset of the complete drive.

Green LED: BUSY
This LED has the same meaning as the (single) LED on commercial drives. It indicates that the drive is busy.

Green LED: OPERATIONAL
This LED is active when the verification drive is in an operational stage. It is turned on as soon as the drive's internal hardware finishes its configuration process successfully. A hardware reset of the drive (e.g. by means of the reset button) will force a new configuration cycle, in that case causing the OPERATIONAL LED to be inactive for a short period of time.

Red LED: DISC DETECTED
This LED indicates (when active) that a supported disc type was detected with success.

Red LED: HIGH GAIN
This LED is lit together with the DISC DETECTED LED whenever the drive's laser is in high gain mode. This high gain mode is (usually) required for reading DVD+RW discs, which have low reflectivity. The drive's processor will put the Basic Engine in high gain mode after two unsuccessful attempts to read a disc's TOC, or when the user forces high gain mode by keeping the EJECT button pressed (while inserting a disc) until the HIGH GAIN LED is lit.

The status of the HIGH GAIN LED is only valid when using a JBE Basic Engine equipped drive. In case of an ASD1 engine based drive this signal does not exist; The ASD1 has automatic gain control.

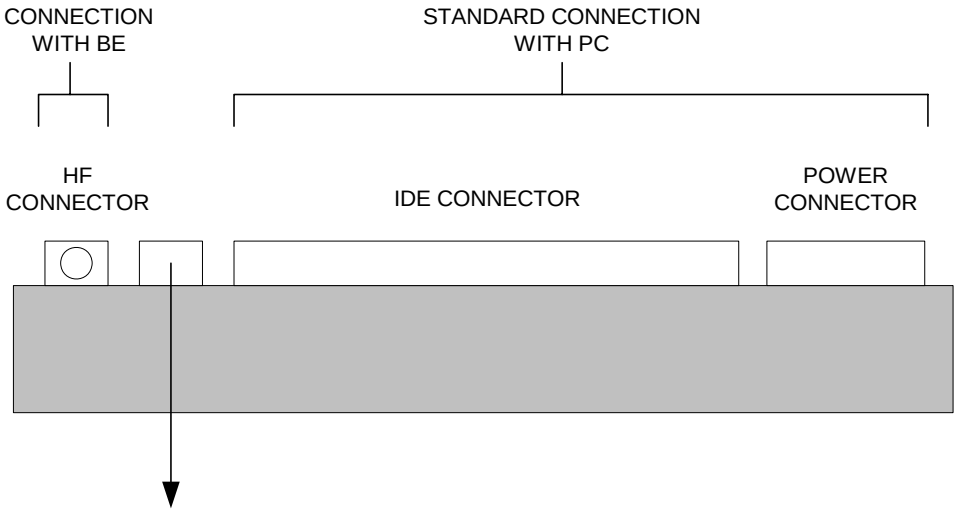
If lit without the DISC DETECTED LED being active, the HIGH GAIN LED indicates that the drive detected an error. This is likely to occur with CD's and damaged discs of any kind.

Red LED: KEY DETECTED
Only relevant for SACD discs.

Red LED: SCRAMBLED
This LED is lit whenever the last sector that passed the drive's head contained scrambled data. This can be any kind of scrambled data. The criterion is the status of the SECB bit corresponding with the least significant bit of the CPS_ID field (Copyright Protection System Identifier), part of the CPSI field (Copyright Protection System Information) of the sector header data.

Connector: I^2C
Not used.

B.5 REAR-SIDE CONNECTORS



MASTER / SLAVE
JUMPER BLOCK

MA = MASTER
SL = SLAVE
CS = CABLE SELECT

IMPORTANT NOTE:
THE VERIFICATION DRIVE WILL ONLY FUNCTION CORRECTLY WHEN CONFIGURED AS MASTER

B.6 SUPPORTED DISC TYPES

The drive has been designed to accept the following disc types:

DVD
- Single layer discs
- Dual layer OTP (Opposite Track Path) discs
- Dual layer PTP (Parallel Track Path) discs

DVD+RW
- Single Layer

➔ CD discs (any kind) are not supported !!!

B.7 READ ERROR BEHAVIOUR

The API interface between the DVD+RW Video Format Verifier tool and the drive has been equipped with some extra functionality to increase the robustness:
When a drive read call has failed or the returned sector data is found to be incorrect a (single) re-read of the sector is done, which is reported by a proper error message.
But often, certainly when it has been directed into the “ice”, an (internal) error condition is generated of which the drive can not escape. In this case the drive has to be reset manually by the user. This is more likely to happen with a JBE based drive than with an ASD1 based drive.

Such read errors may occur because of a number of reasons:
- The disc being read is damaged
- Drive has been exposed to heavy vibration
- At layer jump of a dual layer disc a single read retry can occur
- The drive is trying to read “ice” sectors.